CAN AVR

CAN - grifo® Mini Module AVR

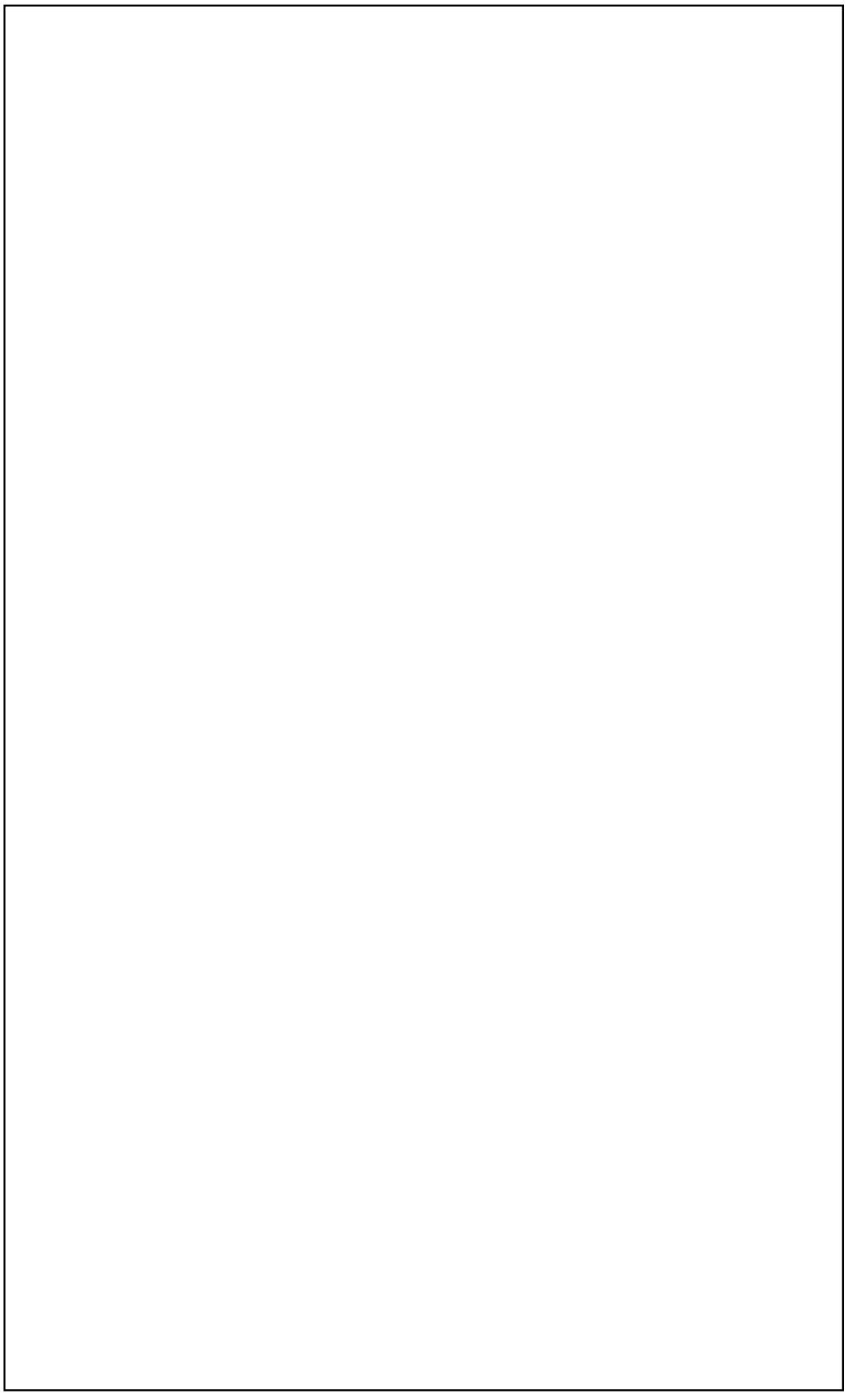
TECHNICAL MANUAL

grifo®
ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661

CAN AVR Rel. 5.00 *Edition 19 January 2006*

abaco, **GPC®**, **grifo®**, *are trade marks of* **grifo®**



CAN AVR

CAN - grifo® Mini Module AVR

TECHNICAL MANUAL

Standard container with **28** pins male socket, dual in line, **100** mils pitch, **600** mils width; very small dimension: 42 x 25 x 15 mm; single power supply voltage required in the wide range from **5.0 Vdc** (the current consumption can change according with module connections); availability of power saving setting as **Idle Mode** and **Power Down Mode**; **Atmel AT90CAN128** microcontroller with external **12 MHz** crystal; **128 KBytes FLASH** for code, **up to 8KBytes FLASH** can be reserved for Boot Loader, **4KBytes RAM** for data, **4KBytes EEPROM** for data; **2** analog comparators channels with different input and output configurations options that allow to create easily bipolar A/D conversion; 5 channels multiplexed **A/D** converter, resolution **10 bits** and **6 µsec** conversion time; **37 Interrupt** sources; **4 Timers Counters**, up to **16** bits, with functions of **PWM**, compare, capture, etc.; up to **24** digital **I/O** lines available on connector. Some of these lines have multiple functions; one hardware serial communication line with programmable physical protocol (Baud Rate up to 115200 Baud, 8 or 9 data bits, 1 or 2 stop bit), **RS 232** buffered or at **TTL** level; one **CAN** differential serial line completely compatible with **CAN 2.0** specifications part **A** and **B**; **JTAG** interface for in-circuit remote debugging; **I²C BUS** controller, completely software configurable; **SPI** interface programmable for syncronous, high speed communications; **Reset** circuitery; one **Real Time Clock** for long and accurate timing, active even in low power modes, capable to manage date and time and to generate periodic interrupts; **240** bytes of SRAM for configuration parameters; **RTC** and **SRAM** backed with on-board **Lithium** battery; **2** status **LED** managed by software through two digital I/O lines; **Boot Loader** preinstalled that allows to program FLASH and EEPROM through **RS 232** serial port; internal FLASH and EEPROM can be managed through **ISP** (**In System Programming**); wide range of Software Development tools used to develop the user application program, as: **Assembler**; **C** compilers (ICC AVR, DDS Micro C); **BASIC Compilers** (BASCOM AVR); etc.; long list of Demo Programs and use examples supplied under source form, duly remarked, and executable form for the available development tools.

grifo®
ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661

CAN AVR Rel. 5.00 *Edition 19 January 2006*

abaco, **GPC®**, **grifo®**, *are trade marks of* **grifo®**

DOCUMENTATION COPYRIGHT BY grifo®, ALL RIGHTS RESERVED

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, either electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written consent of grifo®.

IMPORTANT

Although all the information contained herein have been carefully verified, grifo® assumes no responsability for errors that might appear in this document, or for damage to things or persons resulting from technical errors, omission and improper use of this manual and of the related software and hardware.

grifo® reserves the right to change the contents and form of this document, as well as the features and specification of its products at any time, without prior notice, to obtain always the best product.

For specific informations on the components mounted on the card, please refer to the Data Book of the builder or second sources.

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:

Attention: Generic danger

Attention: High voltage

Attention: ESD sensitive device

Trade Marks

, **GPC®**, grifo® : are trade marks of grifo®.

Other Product and Company names listed, are trade marks of their respective companies.

GENERAL INDEX

INTRODUCTION 1

CARD VERSION 3

GENERAL INFORMATION 4
- DIGITAL I/O LINES 6
- SERIAL COMMUNICATION 6
- CLOCK 6
- MEMORY DEVICES 8
- I²C BUS INTERFACE 8
- SPI INTERFACE 8
- CAN INTERFACE 9
- DIP SWITCH 9
- A/D CONVERTERS 10
- ANALOG COMPARATOR 10
- TIMER COUNTER AND PWM 10
- WATCH DOG 10
- REAL TIME CLOCK 10

TECHNICAL FEATURES 12
- GENERAL FEATURES 12
- PHYSICAL FEATURES 13
- ELECTRIC FEATURES 13

INSTALLATION 14
- VISUAL SIGNALATIONS 14
- CONNECTIONS 14
  - CN1 - EXTERNAL POWER SUPPLY CONNECTOR 14
- MINI MODULE CONFIGURATION 16
- SERIAL COMMUNICATION SELECTION 18
- CONNECTOR SIGNALS INTERFACEMENT 20
- POWER SUPPLY 20
- INTERRUPTS 21
- JTAG INTERFACE 21

SUPPORT CARDS 22
- USE WITH GMB HR84 MODULE 22
- USE WITH CAN GMT MODULE 24

HOW TO START 26
- RICOMPILATION WITH BASCOM AVR 30
- RICOMPILATION WITH ICC AVR 32

SOFTWARE DESCRIPTION 34

PERIPHERAL DEVICES SOFTWARE DESCRIPTION 36
ACTIVITY LEDS 36
DIP SWITCH 36
BACKED SRAM + SERIAL RTC 37
CPU INTERNAL PERIPHERALS 37

BIBLIOGRAPHY 38

APPENDIX A: DATA SHEET A-1
AT90CAN128 ..... A-1
PCF8583 ..... A-2

APPENDIX B: ISP PROGRAMMING WITH GMM TST 2 ..... B-1
COMMON OPERATIONS FOR ALL PROGRAMMERS B-1
PROGRAMMING USING grifo® MP-AVR 51+; grifo® UEP 49 ..... B-4
PROGRAMMING USING EQUINOX EPSILON5 ..... B-7
PROGRAMMING USING ATMEL AVR ISP B-9

APPENDIX C: CAN GMT ELECTRIC DIAGRAM C-1

APPENDIX D: ALPHABETICAL INDEX D-1

FIGURES INDEX

FIGURE 1: LOCATION OF REVISION NUMBER 3
FIGURE 2: BLOCKS DIAGRAM 7
FIGURE 3: CAN AVR PHOTO 9
FIGURE 4: COMPONENTS MAP (COMPONENTS SIDE) 11
FIGURE 5: COMPONENTS MAP (SOLDER SIDE) 11
FIGURE 6: LEDS TABLE 14
FIGURE 7: CN1 - SOCKET WITH MINI MODULE SIGNALS 15
FIGURE 8: DSW2 4 WAYS DIP SWITCH TABLE 16
FIGURE 9: DSW1 8 WAYS DIP SWITCH TABLE 17
FIGURE 10: EXAMPLE OF RS 232 SERIAL CONNECTION 19
FIGURE 11: EXAMPLE OF TTL SERIAL CONNECTION 19
FIGURE 12: LEDS, DIP SWITCHED, ETC. LOCATION 21
FIGURE 13: IMAGE OF MODULE CAN GMT AND CAN AVR 23
FIGURE 14: CONNECTION EXAMPLE OF CAN GMT + CAN AVR 25
FIGURE 15: RS 232 SERIAL CONNECTION BETWEEN A CAN AVR AND A PC 26
FIGURE 16: EXAMPLES TABLE 27
FIGURE 17: BOOT LOADER UTILITY 29
FIGURE 18: LOADING A SOURCE FILE WITH BASCOM AVR 30
FIGURE 19: CONFIGURATION OF COMPILER BASCOM AVR 31
FIGURE 20: COMPILATION WITH BASCOM AVR 31
FIGURE 21: LOADING PROJECT FILE WITH ICC AVR 32
FIGURE 22: COMPILATION WITH ICC AVR 32
FIGURE 23: TOP VIEW AND BOTTOM VIEW OF CAN AVR 35
FIGURE 24: AVAILABLE CONNECTIONS DIAGRAM 39
FIGURE B-1: DIAGMAM OF INTERFACE BETWEEN GMM TST 2 AND MP-AVR 51+ B-3
FIGURE B-2: DIAGMAM OF INTERFACE BETWEEN GMM TST 2 AND UEP 49 ..... B-3
FIGURE B-3: COMPONENT SELECTION USING PG4UW B-4
FIGURE B-4: ISP PROGRAMMING SOCKET FOR CAN AVR AND GMM TST 2 ..... B-5
FIGURE B-5: PROGRAMMER CONFIGURATION USING PG4UW B-6
FIGURE B-6: COMPONENT CONFIGURATION USING PG4UW B-6
FIGURE B-7: PROJECT CONFIGURATION USING EQTOOLS B-8
FIGURE B-8: CPU SELECTION USING AVR STUDIO B-10
FIGURE B-9: CPU CONFIGURATION USING AVR STUDIO B-10
FIGURE B-10: AVR ISP CONFIGURATION USING AVR STUDIO B-10
FIGURE C-1: CAN GMT ELECTRIC DIAGRAM C-1

INTRODUCTION

The use of these devices has turned - IN EXCLUSIVE WAY - to specialized personnel.
This device is not a **safe component** as defined in directive **98-37/CE**.

Pins of Mini Module are not provided with any kind of ESD protection. They are connected directly to their respective pins of microcontroller. Mini Module is affected by electrostatic discharges. Personnel who handles Mini Modules is invited to take all necessary precautions to avoid possible damages caused by electrostatic discharges.

The purpose of this handbook is to give the necessary information to the cognizant and sure use of the products. They are the result of a continual and systematic elaboration of data and technical tests saved and validated from the manufacturer, related to the inside modes of certainty and quality of the information.

The reported data are destined- IN EXCLUSIVE WAY- to specialized users, that can interact with the devices in safety conditions for the persons, for the machine and for the enviroment, impersonating an elementary diagnostic of breakdowns and of malfunction conditions by performing simple functional verify operations , in the height respect of the actual safety and health norms.

The informations for the installation, the assemblage, the dismantlement, the handling, the adjustment, the reparation and the contingent accessories, devices etc. installation are destined - and then executable - always and in exclusive way from specialized warned and educated personnel, or directly from the TECHNICAL AUTHORIZED ASSISTANCE, in the height respect of the manufacturer recommendations and the actual safety and health norms.

The devices can't be used outside a box. The user must always insert the cards in a container that rispect the actual safety normative. The protection of this container is not threshold to the only atmospheric agents, but specially to mechanic, electric, magnetic, etc. ones.

To be on good terms with the products, is necessary guarantee legibility and conservation of the manual, also for future references. In case of deterioration or more easily for technical updates, consult the AUTHORIZED TECHNICAL ASSISTANCE directly.

To prevent problems during card utilization, it is a good practice to read carefully all the informations of this manual. After this reading, the user can use the general index and the alphabetical index, respectly at the begining and at the end of the manual, to find information in a faster and more easy way.

CARD VERSION

This handbook make reference to card **CAN AVR** with printed circuit version **220205**.
The validity of the information contained in this manual is subordinated to the printed circuit revision number and so the user must always check the correspondance.
The printed circuit revision number is always printed in several positions of the circuit and following figure shows the easiest to access.

PRINTED CIRCUIT REVISION NUMBER

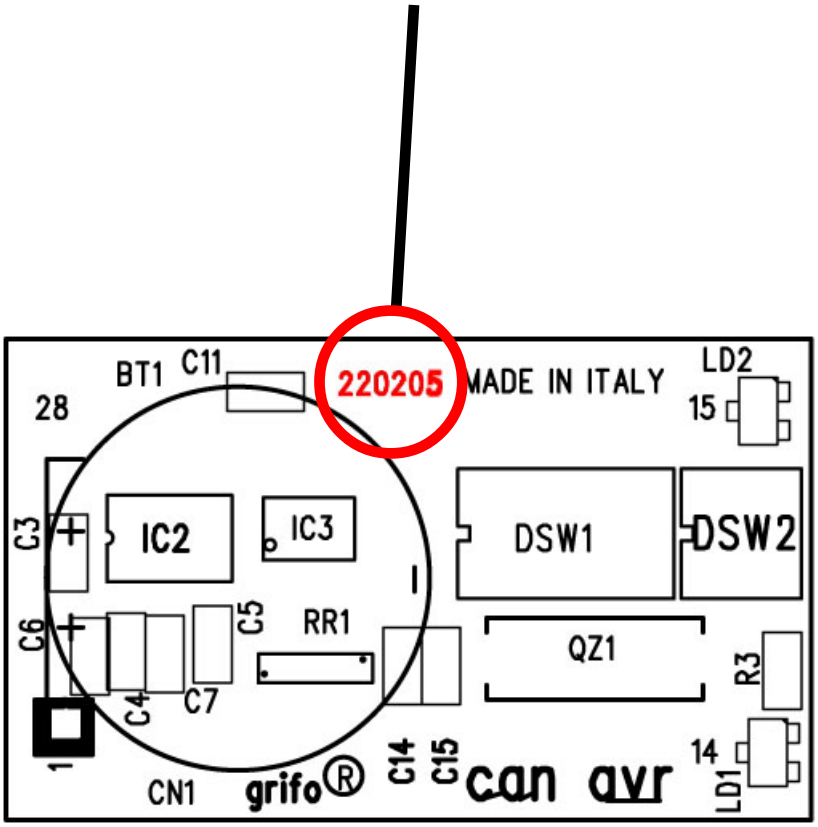


FIGURE 1: LOCATION OF REVISION NUMBER

GENERAL INFORMATION

CAN AVR (CAN - **grifo®** Mini Module AVR) is a module based on microcontroller **Atmel AT90CAN128**, a powerful and complete system on a chip provided with **CPU**, integrated memory (both for data and code), a **watch dog**, many digital I/O lines, 4 multifunction **timers counters** with capture and compare capability, one asynchronous serial line, one CAN serial line, one **I²C BUS** synchronous seria line, 2 comparators for analog signals, etc.
In module's very small area, are already mounted the components that exploit microcontrollers's performance and that allow each functionality mode like a reset generator circuitery.
Possible applications of **CAN AVR** Mini Modules are several.
For example, native CAN application, that is **car automation** (lights turning ON/OFF, heating and cooling systems control, supervision of electric devices, anti-teft and acces control systems, functionality checks, etc.). Also, **connection** on CAN networks with **your own protocols** or with standard protocols (like CANopen, DeviceNet, SDS, CAN Kingdom etc.).
We remark the employ as **smart intelligent nodes** with local functionalities as PID algorithms for controlling temperatures, motors, valves, etc. or as **decentralized systems** as robots, automation of production line machines, big factory automations.
Finally, **teleacquisition** and **telecontrol** on medium and low distances, **conversion** between **CAN** and **asynchronous serial** line or **I²C BUS** line and **home automation** (lights turning ON/OFF, heating and cooling systems control, supervision of electric devices, security and acces control systems).
It is really important the **Didactics** use in fact **CAN AVR** offers a very low cost system suitable to learn a **RISC** microcontroller with AVR core and to develop the typical start applications for the students.
For this purpose it is likewise interesting the **CAN GMT** support card that solves the problems for power supply, RS 232 serial connection to development PC, CAN connection and for module lines connection.
CAN AVR is delivered with a Boot Loader preinstalled. This program allows to reprogram Flash and EEPROM through a simple RS 232 serial port (for example, PC COM port).
On board microcontroller electronics also supports the well-known **JTAG** interface, that allows to control its working status directly on the application it is running.
In this case debugging is controlled through the PC and a specific interface to connect to the six signals of Mini Module directly available on its socket.
The PC program that controls JTAG interface is AVR Studio, free IDE developed by Atmel they distribute through their website.
Both the solutions ensure a short time to market: the user can obtain a prototype or even a ready product **in one week**.
Overall features are:

- Standard container with **28** pins male **socket**, dual in line, **100** mils pitch, 600 mils width
- Very small dimension: 42 x 25 x 15 mm
- Single power supply voltage required in the wide range from **+5.0 Vdc** (the current consumption can change according with module connections)
- Availability of power saving setting as **Idle Mode** and **Power Down Mode**
- **Atmel AT90CAN128** microcontroller with external **12 Mhz** crystal
- **128 KBytes FLASH** for code, **up to 8KBytes FLASH** can be reserved for Boot Loader, **4KBytes RAM** for data, **4KBytes EEPROM** for data

- 2 analog comparators channels with different input and output configurations options that allow to create easily bipolar A/D conversion
- 5 channels multiplexed **A/D** converter, resolution **10 bits** and **6 μsec** conversion time
- 37 **interrupt** sources
- **4 Timers Counters**, up to 16 bits, with functions of **PWM**, compare, capture, etc.
- Up to **24** digital **I/O** lines available on connector (some of these lines have multiple functions)
- One hardware serial communication line with programmable physical protocol (Baud Rate up to 115200 Baud, 8 or 9 data bits, 1 or 2 stop bit), **RS 232** buffered or at **TTL** level
- One **CAN** differential serial line completely compatible with **CAN 2.0** specifications part **A** and **B**
- **JTAG** interface for in-circuit remote debugging
- **I²C BUS** controller, completely software configurable
- **SPI** interface programmable for syncronous, high speed communications
- **Reset** circuitery
- One **Real Time Clock** for long and accurate timing, active even in low power modes, capable to manage date and time and to generate periodic interrupts
- **240** bytes of SRAM for configuration parameters
- **RTC** and **SRAM** backed with on-board **Lithium** battery
- **2** status **LED** managed by software through two digital I/O lines
- **Boot Loader** preinstalled that allows to program FLASH and EEPROM through RS 232 serial port
- Internal FLASH and EEPROM can be managed through **ISP** (**In System Programming**)
- Wide range of Software Development tools used to develop the user application program, as: **Assembler**; **C** compilers (ICC AVR, DDS Micro C); **BASIC Compilers** (BASCOM AVR); etc
- Long list of Demo Programs and use examples supplied under source form, duly remarked, and executable form for the available development tools.

Here follows a description of the board's functional blocks, with an indication of the operations performed by each one.
To easily locate such section on verify their connections please refer to figure 2.

DIGITAL I/O LINES

The Mini Module **CAN AVR** is provided with 24 TTL digital I/O lines, of the microprocessor Atmel AT90CAN128, that are all the signals of Port PB, signals from 0 to 6 of PE, signals 0, 1, 4 and 7 of PD and signals 0 an from 4 to 7 of PF.
These lines are connected directly to 28 pins connectors with standard **grifo®** Mini Module pin out, allowing to be connected direclty to several interface cards.
By software it is possible to define and acquire the function and the status of these lines, and also to match them to peripheral devices (like Timer Counter, Interrupt, I^2C BUS, SPI, etc.), simply programming some CPU internal registers.
For further information please refer to paragraph CONNECTIONS and PERIPHERAL DEVICES SOFTWARE DESCRIPTION.

SERIAL COMMUNICATION

On **CAN AVR** there is always availability of one hardware serial line that is completely software configurable for physical protocol (baud rate, stop bits number, lenght of character, etc) by simply programming some microprocessor registers as described in the manufacturer documentation or in the appendix of this manual.
The serial lines are connected to CN1 connector at TTL or RS 232 level, thanks to some on board dip switches configuration, so when the card must be connected in a network or at long distance or with other systems that use different electric protocol, the user must provide external drivers (RS 232, RS 422, RS 485, Current loop, etc.). Please remember that on CN1 connector more than standard receive and transmit signals are available also other I/O signals that can be driven by software; these signals can be used to define the RS 485 line direction, to enable the RS 422 transmit drive or to generate an RS 232 handshake. For example it can be used the **MSI 01** module that converts a TTL serial line in any other electric standards in a pratical and inexpensive way.
Please read SERIAL COMMUNICATION SELECTION paragraph of this manual or contact directly **grifo®** technician for further explanation or any other necessary information.

CLOCK

On **CAN AVR** module there are two separate and indipendent circuits based on crystals, that generate the clock signals for the microcontroller and the RTC module.
The first generates a 12 MHz while the second generates a 32768 Hz.
The choice of using two circuits and two separated clock sources, has the advantage to reduce cost in the larger number of low, middle speed applications and to afford the high speed applications when necessary.
About speed and performances please remind that **CAN AVR** has a RISC microcontroller on board, capable to execute in average one instruction per clock cycle.
So, considering the frequency of quartz installed, execution speed may be almost 12 MIPS.

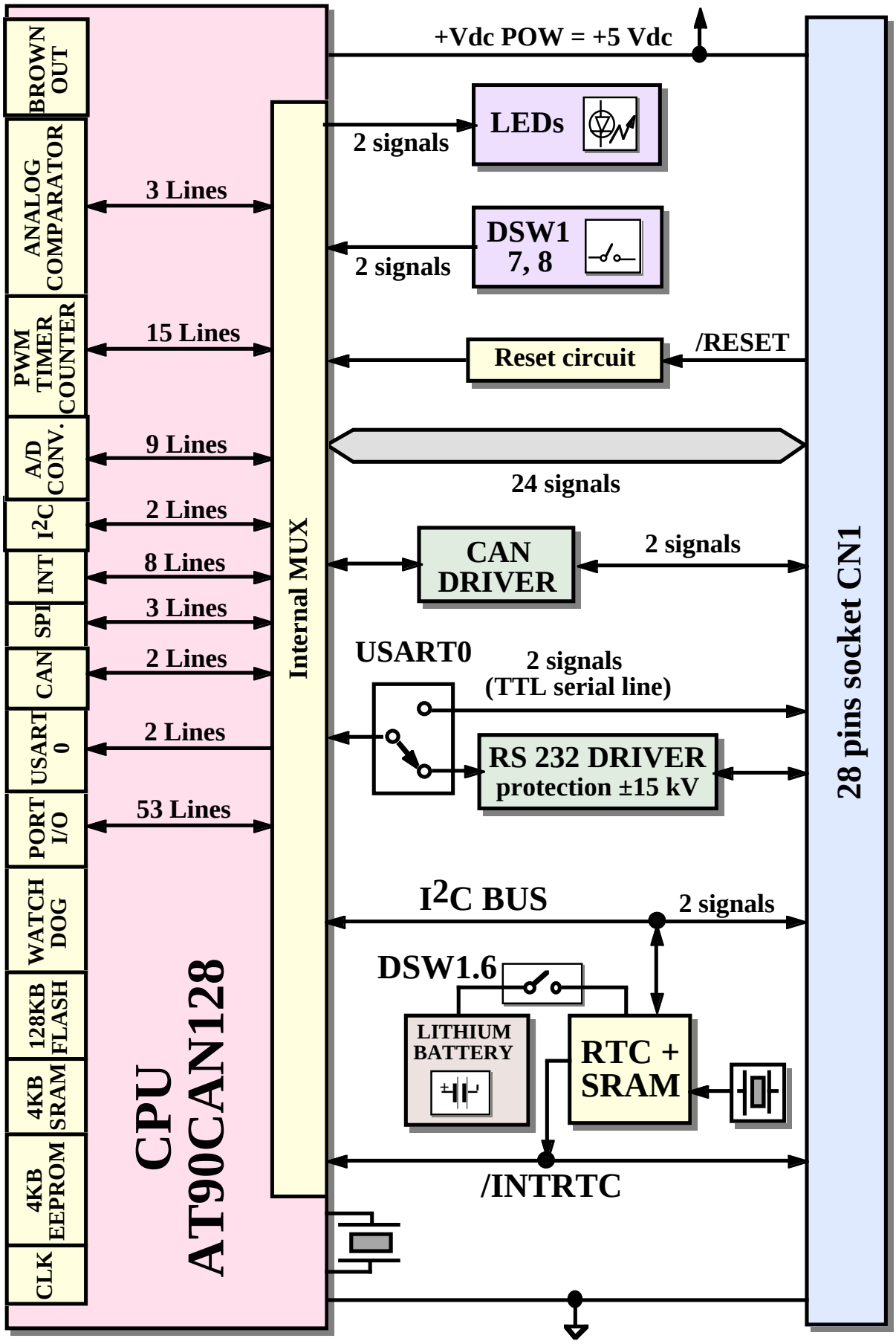


FIGURE 2: BLOCKS DIAGRAM

MEMORY DEVICES

The card is provided of 136.25K of memory divided with a maximum of **128K** Bytes FLASH EPROM, **4KBytes** Bytes of internal SRAM, **4KBytes** EEPROM and 240 Bytes in RTC module, reachable only through I²C BUS.
The memory configuration must be chosen considering the application to realize or the specific requirements of the user.
Thanks to on board EEPROM (and SRAM of RTC when the back up battery is connected) there is the possibility to keep data also when power supply is failed.
In this way the card is always able to maintain parameters, logged data, system status and configuration, etc. in each working conditions.
Whenever the amount of memory for data is not sufficient (i.e. for data loghin systems), it is always possible to connect external memory devices (with SRAM, EEPROM, FLASH technologies) through the comfortable and efficient SPI and I²C BUS interface of the card.
The addressing of memory devices is controlled by microcontroller as described in the component data sheet or in APPENDIX A of this manual.

I²C BUS INTERFACE

Standard pin out of 28 pins **grifo®** Mini Module connector reserves two pins, 6 and 7, to I²C BUS interface. These signals are provided with a 4.7 kΩ pull-up on the Mini Module board.
CAN AVR features a hardware I²C BUS interface and is managed through microcontroller inernal registers. It can work both as master or slave, in transmission or reception.
This interface allows to connect components featuring the same communication standard to expand locally the potentialities of Mini Module.
For this Mini Module, a rich serie of demo programs that explain how to use A/D and D/A converters, memories, temperature sensors, on-board RTC, etc. by a complete and well commented code is available.
Remarkable is the possibility to connect the **grifo®** **QTP** operator panels serie through I²C BUS. **QTP** are capable to manage alphanumeric and graphicdisplay and several models of keyboards, according to the model selected.
Mini Modules support cards manufactured by **grifo®** (like **CAN GMT**) are provided also with a connector dedicated to I²C BUS, to easy the field connections.
Please remark that RTC with backed SRAM permanently takes the slave address **A0**, so user applications cannot use it.
For further information please refer to component data sheet or appendix A of this manual.

SPI INTERFACE

Mini Module **grifo®** **CAN AVR** is provided with a SPI serial inteface featured through a specific hardware section of the microcontroller.
Signals MISO, MOSI and SCKof SPI interface are available respectively on pins 8, 9 and 12 of connector CN1.
All interface parameters are managed through microcontroller inernal registers.
For further information please refer to component data sheet or appendix A of this manual.

CAN INTERFACE

Mini Module **CAN AVR** is provided with a powerful hardware CAN inbterface, capable to reach the maximum bit rate of 1 Mbp/sec.Signals CAN L and CAN H are available respectively on pins 8 and 9 of CN1.
This interface is fully compatible with CAN 2.0 specifications parts A and B.

DIP SWITCH

CAN AVR Mini Module is provided with two dip switches (ab eight ways and a four ways) whose purpose is to set up several electric parameters of module itself and the card operating mode.
In fact DSW1 allows to decide whether serial interface **USART0** must be buffered as **RS 232** or **TTL**, in this latter case allowing to use and external buffer like **RS 422**, **RS 485** or **current loop** drivers.
Four ways DSW2 allows to decide whether the signals of CAN interface or the signals of SPI interface must be available on pins 8 and 9 of CN1.
DSW1 also allows to connect or not the back up battery of RTC + SRAM module.
In addition, status of some swithces can be acquired by software through reading specific microcontroller signals (PA2 and PA3).
For further information please see also the paragraph MINI MODULE CONFIGURATION.

FIGURE 3: CAN AVR PHOTO

A/D CONVERTERS

Mini Module **CAN AVR** is provided with a five channels A/D converter, multiplexed on signals of port PF, with 10 bit of resolution.
Reference voltage can be fetched externally through a pin or internal reference generator can be used.
Range of connectable signals is from 0 to +Vdc POW, analog signal generator must have a low impedance.
Conversion end can be used to trigger an interrupt, if enabled.
Peripheral management is performed by manipulating specific microcontroller internal registers.
For further information please refer to Appendix A or to comments in source code of our examples.

ANALOG COMPARATOR

Microcontroller Atmel AT90CAN128 features an analog comparator that can select as input both an internal reference voltage and several pins (pin 10 and 11 of CN1, but also A/D converter inputs) through an internal multiplexer.
For further information, please refer to application notes on Atmel web site.

TIMER COUNTER AND PWM

Microcontroller is provided with four Timer/Counter (two featuring eight bits, two featuring sixteen bits) capable to count clock pulses (through a programmable prescaler), level transictions on specific pins and to generate interrupts.
They can also be used in PWM mode, to generatesignals of variable duty cycle and frequency set by software with eight or sixteen bits of resolution.
Typical applications of these signals are motor velocity control, in fact several motor control cards are provided with compatible inputs.
Another application is analog signals generation by simply adding an integrator.

WATCH DOG

Microcontroller Atmel AT90CAN128 features a hardware programmable watch dog capable to reset the CPU if it is enabled and the application program does not retrigger it in the activation time.
This latter time can be programmed between 16 msec and 2 seconds.

REAL TIME CLOCK

Mini Module features an on-board Real Time Clock (in I^2C BUS with slave address **A0**), capable to manage hours, minutes, seconds, year, month, day and weekday. It also features 240 bytes of SRAM
This component can be completely managed by software and is backed through a back up battery that, if enabled, can warrant data validity in any operating condition.

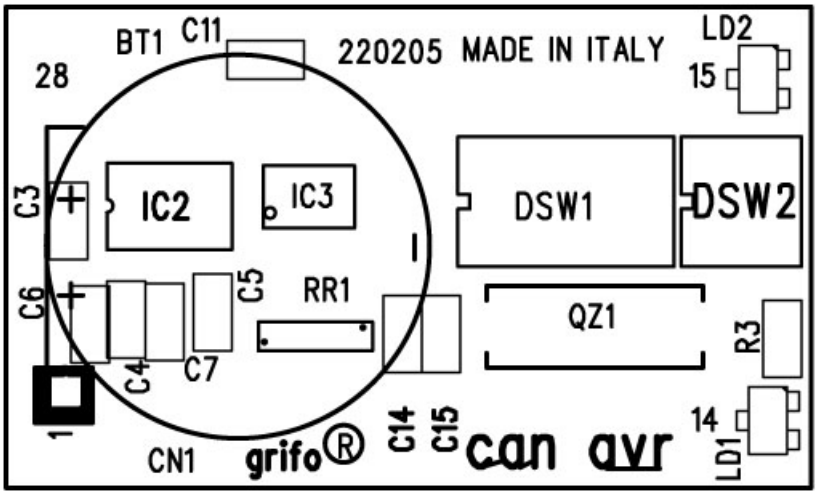


FIGURE 4: COMPONENTS MAP (COMPONENTS SIDE)

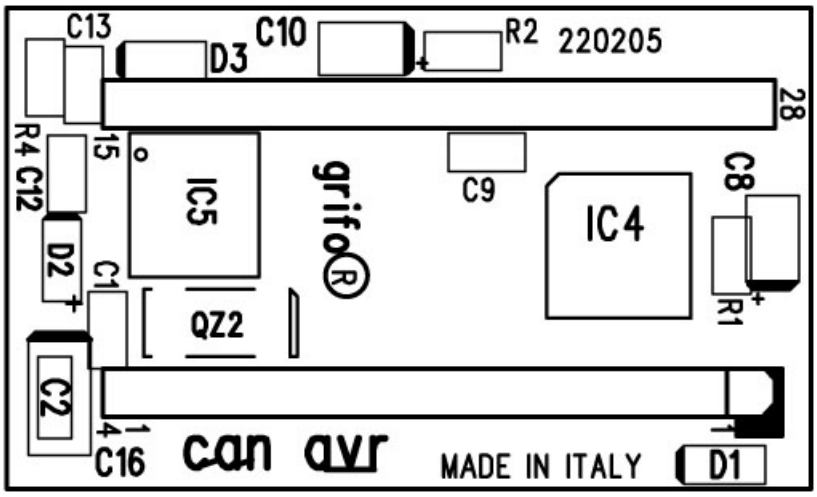


FIGURE 5: COMPONENTS MAP (SOLDER SIDE)

GENERAL FEATURES

Resources:	24 TTL digital I/O lines 2 analog inputs on comparator 5 channels A/D converter 4 Timer/Counter for compare, capture, PWM 1 Watch dog section 1 SPI interface 1 I^2C BUS interface 1 CAN interface 1 Real Time Clock section 37 interrupt sources 1 hardware serial line RS 232 or TTL 1 Dip switch 8 ways, 1 Dip switch 4 ways 2 status LEDs
Memories:	128 KBytes FLASH EPROM for code 4 KBytes SRAM for data 4 KBytes EEPROM for data 240 Bytes external SRAM for data (on I^2C BUS)
CPU:	Atmel AT90CAN128
Clock frequency:	12 MHz
Counter maximum frequency:	Clock I/O frequency
Power on time:	from 37 msec to 93 msec, settable through configuration bits
Watch dog intervent time:	programmable from about 16 msec up to 2 sec
A/D converter resolution:	10 bits
A/D conversion time:	min. 6 μsec (for 10 bits)

PHYSICAL FEATURES

Size: (W x H x D):	42 x 25 x 15 mm
Weight:	12 g
Connectors:	28 pins male socket DIL, 100 mils pitch, 600 mils width
Temperature range:	0÷50 °C
Relative humidity:	20%÷90% (without condense)

ELECTRIC FEATURES

Power supply voltage:	+Vdc POW = +5.0 Vdc
Current consumption:	
	minimum: 12 mA
	normal: 31 mA
	maximum: 40 mA
Back up battery voltage:	3.0 Vdc
Back up battery consumption:	3.9 µA
Impedance analog signals generators:	<10 kΩ
RS 232 protection:	±15 kVdc
I²C BUS pull-up resistor:	4.7 kΩ
Brown out threshold:	programmable from 2.5 to 4.1 Vdc, with hysteresis

INSTALLATION

In this chapter there are the information for a right installation and correct use of the **CAN AVR** card. In detail there are the locations and functions of each connector, of the user settable dip switches, LEDs, and so on.

VISUAL SIGNALATIONS

CAN AVR features the LEDs described in the following table:

LED	COLOUR	PURPOSE
LD1	Green	Driven by signal PA.0 , AD0 of Mini Module, it can be used as activity LED and can be managed by software.
LD2	Red	Driven by signal PA.1 , AD1 of Mini Module, it can be used as activity LED and can be managed by software.

FIGURE 6: LEDS TABLE

The main function of LEDs is to inform the user about card status, with a simple visual indication and in addition to this, LEDs make easier the debug and test operations of the complete system. To recognize the LEDs location on the card, please refer to figure 12. while for further information please refer to paragraph ACTIVITY LEDS.

CONNECTIONS

The **CAN AVR** module has 1 connector that can be linkeded to other devices or directly to the field, according to system requirements.
In this paragraph there are connector pin out, a short signals description (including the signals direction) and connectors location (see figure 12) that simplify and speed the installation phase. Some additional figures shows the pins functionalities and some of the most frequently used connections.

CN1 - EXTERNAL POWER SUPPLY CONNECTOR

CN1 is a 28 pins, male, dual in line, socket connector with 100 mils pitch and 600 mils width.
On CN1 are available all the interfacement signals of the Mini Module as the power supply, the I/O lines, the synchronous and asynchronous communication lines, the on board peripheral devices signals, the operating mode selection lines, etc.
Some pins of this connector have multiple purposes, in fact they can be multiplexed by programming some software registers with several CPU internal devices and the following figure lists all these possible functionalities. So the signals available on CN1 have different types as described in the following CONNECTOR SIGNALS INTERFACEMENT paragraph and they follow grifo® Mini Module standard pin out.

To avoid problems in pin counting and numbers the figure 7 shows the signals directly on the top view of the **CAN AVR**; moreover the serigraph reports the pins number on the four corner of the card both on bottom (solder) and top (component) side.

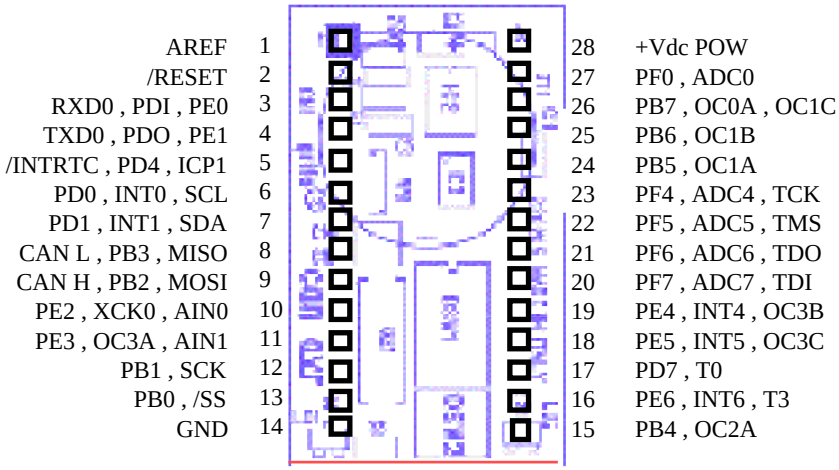

AREF 1
/RESET 2
RXD0 , PDI , PE0 3
TXD0 , PDO , PE1 4
/INTRTC , PD4 , ICP1 5
PD0 , INT0 , SCL 6
PD1 , INT1 , SDA 7
CAN L , PB3 , MISO 8
CAN H , PB2 , MOSI 9
PE2 , XCK0 , AIN0 10
PE3 , OC3A , AIN1 11
PB1 , SCK 12
PB0 , /SS 13
GND 14
28 +Vdc POW
27 PF0 , ADC0
26 PB7 , OC0A , OC1C
25 PB6 , OC1B
24 PB5 , OC1A
23 PF4 , ADC4 , TCK
22 PF5 , ADC5 , TMS
21 PF6 , ADC6 , TDO
20 PF7 , ADC7 , TDI
19 PE4 , INT4 , OC3B
18 PE5 , INT5 , OC3C
17 PD7 , T0
16 PE6 , INT6 , T3
15 PB4 , OC2A

FIGURE 7: CN1 - SOCKET WITH MINI MODULE SIGNALS

Signals description:

RXD0 = I - Receive Data of USART0; it can be buffered as RS 232 or TTL
TXD0 = O - Transmit Data of USART0; it can be buffered as RS 232 or TTL
CAN L = I/O - Bidirectional differatial L signal of CAN interface
CAN H = I/O - Bidirectional differatial H signal of CAN interface
PDI = I - ISP programming data output line
PDO = O - ISP programming data input line
PAX, PBX, ... ,PFx = I/O - CPU I/O TTL signals
SCL = I/O - I^2C Bus clock signal
SDA = I/O - I^2C Bus reception and transmission signal
Tn = I - External inputs for counters of timer n-th
INT n = I - CPU n-th interrupt signal
ADCn = I - N-th A/D converter analog input
AINn = I - N-th comparator analog inputs
AREF = I - A/D converter reference voltage
OCn(A/B/C) = O - Compare output of n-th section
TDO, TDI = I/O - Data signals of JTAG interface
TCK, TMS = I - Clock and control signals of JTAG interface
XCKn = I - Exeranl clock signals for USARTn (in synchronous mode)
/INTRTC = O - Periodic interrupt signal generated by RTC PCF 8583
/RESET = I - Reset signal
MOSI = O - SPI interface data output
MISO = I - SPI interface data input
/SS = I - SPI interface slave select
SCK = I/O - SPI interface clock
+Vdc POW = I - Power supply
GND = - Ground

MINI MODULE CONFIGURATION

On **CAN AVR** module there are an 8 ways dip switch and a 4 ways dip switch that define some configurations of the card. In the following figures is reported their list, their position and their functions in all the available connection modes.
The ***** (asterisk) denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the user receives.
To recognize the configuration elements location, please refer to figure 12.
For further information about serial communication lines, please refer to paragraph SERIAL COMMUNICATION SELECTION.

SWITCH	POSITION	PURPOSE	DEF.
DSW2.1	ON	It connects output CAN L of CAN driver to signal CANL , PB3 , MISO of CN1. Used in conjunction with DSW2.3.	*
	OFF	It does not connect output CAN L of CAN driver to signal CANL , PB3 , MISO of CN1. Used in conjunction with DSW2.3.	
DSW2.2	ON	It connects output CAN H of CAN driver to signal CANH , PB2 , MOSI of CN1. Used in conjunction with DSW2.4.	*
	OFF	It does not connect output CAN H of CAN driver to signal CANH , PB2 , MOSI of CN1. Used in conjunction with DSW2.4	
DSW2.3	ON	It connects signal CANL , PB3 , MISO of CN1 to pin 13 of microcontroller (signal MISO , PB3). Used in conjunction with DSW2.1.	
	OFF	It does not connect signal CANL , PB3 , MISO of CN1 to pin 13 of microcontroller (signal MISO , PB3). Used in conjunction with DSW2.1.	*
DSW2.4	ON	It connects signal CANH , PB2 , MOSI of CN1 to pin 12 of microcontroller (signal MOSI , PB2). Used in conjunction with DSW2.2	
	OFF	It does not connect signal CANH , PB2 , MOSI of CN1 to pin 12 of microcontroller (signal MOSI , PB2). Used in conjunction with DSW2.2	*

FIGURE 8: DSW2 4 WAYS DIP SWITCH TABLE

SWITCH	POSITION	PURPOSE	DEF.
DSW1.1	ON	It connects the serial receive signal RXD0 , PDI , PE.0 of the microcontroller to RS232 driver. Used in conjunction with DSW1.3,5.	*
	OFF	It does not connect the serial receive signal of the microcontroller to RS232 driver, thus allowing the direct connection to RXD0 , PDI , PE.0 signal on CN1. Used in conjunction with DSW1.3,5.	
DSW1.2	ON	It connects TXD0 , PDO , PE.1 signal on CN1 to RS232 serial driver. Used in conjunction with DSW1.4.	*
	OFF	It does not connect TXD0 , PDO , PE.1 signal on CN1 to RS232 serial driver thus allowing the direct connection to microcontroller. Used in conjunction with DSW1.4.	
DSW1.3	ON	It connects RXD0 , PDI , PE.0 signal on CN1 to RS232 serial driver. Used in conjunction with DSW1.1,5.	*
	OFF	It does not connect RXD0 , PDI , PE.0 su CN1 signal on CN1 to RS232 serial driver thus allowing the direct connection to microcontroller. Used in conjunction with DSW1.1,5.	
DSW1.4	ON	It connects TXD0 , PDO , PE.1 signal on CN1 directly to microcontroller, with no use of RS232 serial driver. Used in conjunction with DSW1.2.	
	OFF	It does not connect TXD0 , PDO , PE.1 signal on CN1 to microcontroller, thus allowing the use of RS232 serial driver. Used in conjunction with DSW1.2.	*
DSW1.5	ON	It connects RXD0 , PDI , PE.0 signal on CN1 directly to microcontroller, with no use of RS232 serial driver. Used in conjunction with DSW1.1,3.	
	OFF	It does not connect RXD0 , PDI , PE.0 signal on CN1 to microcontroller, thus allowing the use of RS232 serial driver. Used in conjunction with DSW1.1,3.	*
DSW1.6	ON	It connects on-board battery to RTC PCF 8583, allowing to keep date, time and SRAM content even when power supply is turned off.	
	OFF	It does not connect on-board battery to RTC PCF 8583, so date, time and SRAM content are lost when power supply is turned off.	*
DSW1.7	ON	It connects PA.2 , AD2 signal to GND signal. This switch can be a user input.	
	OFF	It does not connect PA.2 , AD2 signal to GND signal. PA.2, AD2 is connected to +Vdc POW through a pull-up. This switch can be a user input.	*
DSW1.8	ON	It connects PA.3 , AD3 signal to GND signal. This switch can be a user input.	
	OFF	It does not connect PA.3 , AD3 signal to GND signal. PA.3, AD3 is connected to +Vdc POW through a pull-up. This switch can be a user input.	*

FIGURE 9: DSW1 8 WAYS DIP SWITCH TABLE

SERIAL COMMUNICATION SELECTION

Serial communication line of **CAN AVR** can be buffered as RS 232 or TTL.
By software, it is possible to define physical communicatin protocol for the line setting some microcontroller internal registers.
The serial interface has its own group of registers for configuration, and can work in total independence respect to the other peripherals.
Electric protocol is selected by hardware and requires the dip switches to be configured correctly, as described in previous tables; the user can set any configuration in autonomy following the below reported information:

- SERIAL LINE USART0 CONFIGURED AS RS 232 (default configuration)

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	ON
DSW1.4	=	OFF
DSW1.5	=	OFF

- SERIAL LINE USART0 CONFIGURED AS TTL

DSW1.1	=	OFF
DSW1.2	=	OFF
DSW1.3	=	OFF
DSW1.4	=	ON
DSW1.5	=	ON

Figures 10 and 11 show how to connect a generic external system to both serial lines of **CAN AVR**.

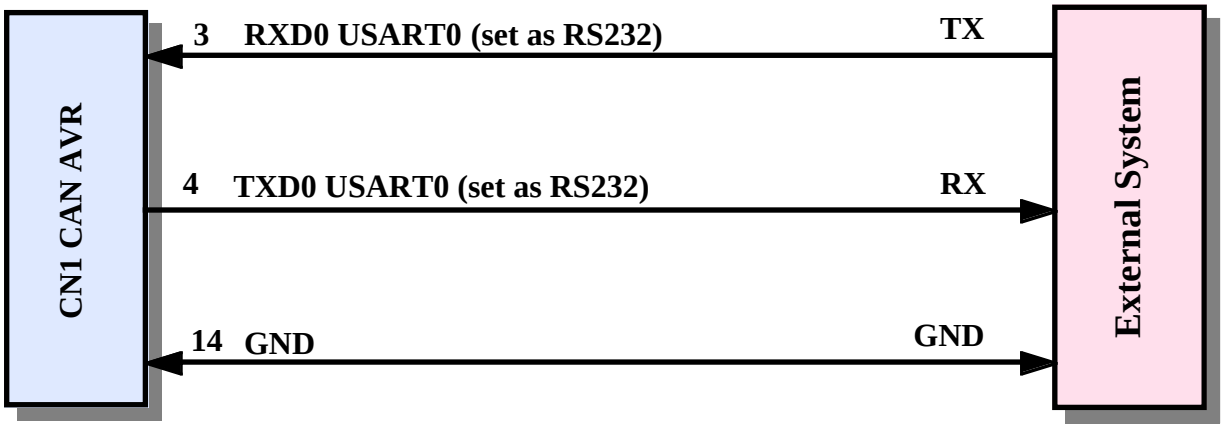


FIGURE 10: EXAMPLE OF RS 232 SERIAL CONNECTION

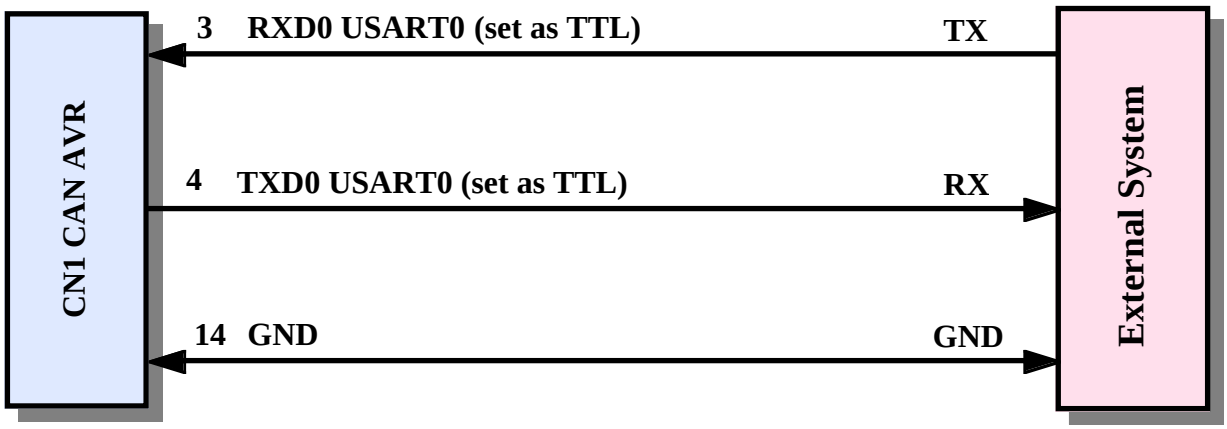


FIGURE 11: EXAMPLE OF TTL SERIAL CONNECTION

CONNECTOR SIGNALS INTERFACEMENT

To prevent possible connecting problems between **CAN AVR** and the external systems, the user has to read carefully the previous paragraph information and he must follow these instrunctions:

- For RS 232 and CAN signals the user must follow the standard specifications of these protocols, defined by specific normatives.

- All TTL signals must follow the rules of this electric standard. The connected digital signals must be always referred to card ground (GND) and then the 0V level corresponds to logic state 0, while the +Vdc POW level corrisponds to logic state 1. The connection of these lines to devices of the controlled system (encoders, switches, proximity, electric valves, power relays, etc.) must be performed through proper power interfaces; it is preferible to adopt opto coupled interfaces that ensure an electric insulation between Mini Module electronic and external noisy, typically generated by power electronic.

- The inputs for analog comparators must be connected to signals generators featuring a low impedance in the range from 0 to +Vdc POW , to assure greater stability and precision.

- The inputs for A/D converter must be connected to signals generators featuring a low impedance in the range from 0 to +Vdc POW, to assure greater stability and precision.

- PWM signals generated by Timer Counter and OCM sections are TTL type so they must be buffered to interface the power circuitery. Typical interfaces can be current driver (if PWM signal is still required) or an intergrator circuit if analog voltage is required.

- Also I^2C BUS and SPI signals are at TTL level, as defined by the same standards; for completeness it is remarked that in a network with several devices and rather long it is better to study the connection lay out and to set properly the output stage, the best operational modes and the programmable bit rate: all these conditions allow communications in any condition. On Mini Module, signals SDA and SCL are pulled-up to +Vdc POW through 4.7 kΩ resistors.

POWER SUPPLY

Mini Module can be supplied by a tension of +5 Vdc called +Vdc POW in this manual.
CAN AVR design adopted all the circuital and componentistic options that reduce sensibility to noise and reduce consumption, including the possibility to switch the microcontroller to low consumption modes.
In optimal situation, the minimun consumption (in power down mode) is lower tham 1 mA, so it can, for example, increase battery life in case of portable applications.
For further information please refer to paragraph ELECTRIC FEATURES.

INTERRUPTS

A remarkable feature of **CAN AVR** card is the powerful interrupt management.
Here follows a short description of which devices can geneate interrupts and their modalities; for further information about interrputs management please refer to the microprocessor data sheet or APPENDIX A of this manual.

- Pin 6 of CN1 -> Generates an interrupt INT0 of microprocessor.
- Pin 7 of CN1 -> Generates an interrupt INT1 of microprocessor.
- Pin 19 of CN1 -> Generates an interrupt INT4 of microprocessor.
- Pin 18 of CN1 -> Generates an interrupt INT5 of microprocessor.
- Pin 16 of CN1 -> Generates an interrupt INT6 of microprocessor.
- CPU peripherals -> Generate an internal interrupt. In detail the possible microcontroller interrupt sources are: Timer Counter, OCM, USART0, CAN, analog comparators, A/D converter, I^2C BUS, SPI, EEPROM.

An interrupt management section, integrated in microcontroller, allows to enable, disable and mask so the user has the possibility to respond promptly and efficently to any external event.
The microcontroller has an interrupt section that let the user manage the 37 interrupt sources. So theapplication program has always the possibility to react promptly to every event.

JTAG INTERFACE

The microcontroller on board of **CAN AVR** is provided with JTAG interface.
Signals of this interface (TDI, TDO, TMS and TCK) are available respectively on pins from 20 to 23 of CN1. Configuration registers of microcontroller must be carefully programmed to enable or disable JTAG functions.

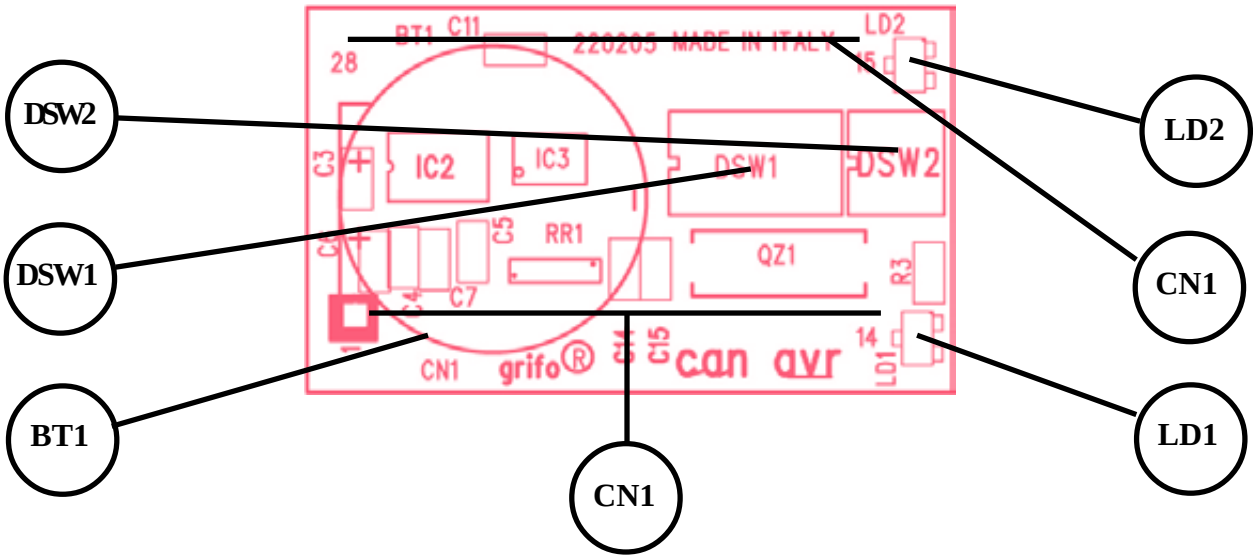


FIGURE 12: LEDS, DIP SWITCHED, ETC. LOCATION

SUPPORT CARDS

CAN AVR Mini Module can be used as a macro components for some support cards either developed by the user or directly chosen from the **grifo®** boards. In the following paragraphs are reported the suggested configuration of the most interesting support cards.

USE WITH GMB HR84 MODULE

Amongst **grifo®** cards, **GMB HR84** module is the one designed specifically to provide to 28 pins Mini Modules many interesting features as: 8 optocoupled inputs, 4 relay outputs, mechanical mounting on omega rails and a comfortable wiring through screw terminal connectors. The complete description of the product is available in the relative data sheet and technical manual while in this paragraph are listed the advantages obtained by using this pair of cards:

GMB HR84 allows easily to:

- to supply the Mini Module through on board power supply;

- to have eight TTL I/O signals of microprocessor ports optocoupled NPN and PNP at the same time and visualized through green LEDs; I/O signals are multiplexed with timer inputs, so developed functions like counters are immediatly available;

- to have four TTL I/O signals of microprocessor ports on bufferd relays driving and visualized through red LEDs;

- to connect on **I^2C BUS** and +5 Vdc power supply on a dedicated connector;

- to connect immediatly communication serial line through a comfortable 9 ways DB9 connector;

- to buffer easily TTL UART0 signals from microprocessor in RS 422, RS 485 or current loop;

- to connect PWM signal through a comfortable standard AMP connector;

The serial connection cable with development PC is the CCR 9+9 R (or in other words a reversed extension cable provided of D9 Female and D9 Male connectors).

FIGURE 13: IMAGE OF MODULE GMB HR 84 AND CAN AVR

USE WITH GMM TST2 MODULE

Amongst **grifo®** cards, **CAN GMT** is the one designed specifically to be the prototyping board supporting **CAN GMx** 28 pins Mini Modules.
CAN GMT allows easily to:

- to supply the Mini Module through on board power supply
- to have I/O port and A/D converter signals on a comfortable low profile connector compiant to standard **I/O ABACO®**
- to have I^2C Bus and interrupt signals on a specific connector, to be able to expand the system with any I^2C Bus device, driven both in polling and in interrupt
- to connect immediatly RS 232, TTL and CAN signals through two comfortalbe D type connectors
- to set and visualize the status of up to 6 microcontroller I/O signals through coloured push buttons and LEDs excludible by jumpers
- to generate sound feedback using the autoscillating on board buzzer
- to develop quickly and comfortably any application taking advantage of the wide prototyping area provided with duplicated signals

The following configuration is suggested to use the couple **CAN GMT** + **CAN AVR** in their base version, that is RUN mode with serial line buffered in RS 232:

*Configuration **CAN AVR***

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	ON
DSW1.4	=	OFF
DSW1.5	=	OFF
DSW1.6	=	OFF
DSW1.7	=	OFF
DSW1.8	=	OFF

*Configuration **CAN GMT***

J1	=	1-2
J2	=	1-2
J3	=	1-2
J4	=	1-2
J5	=	1-2
J6	=	1-2
J7	=	1-2
J8	=	not connected
J9	=	1-2

The serial connection cable with development P.C. is the CCR 9+9 E (or in other words an extension cable provided of D9 Female and D9 Male connectors).

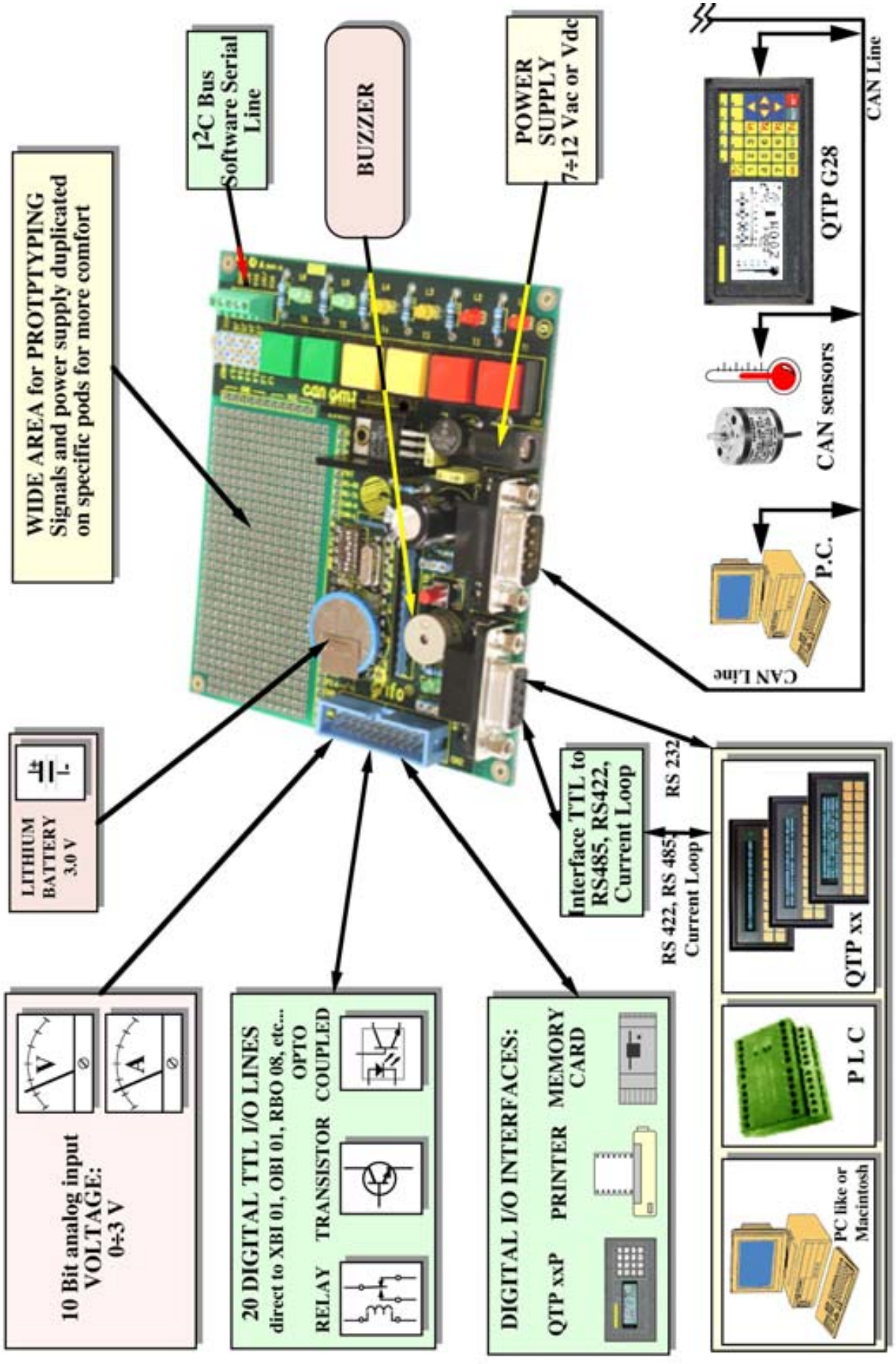


FIGURE 14: CONNECTION EXAMPLE OF CAN GMT + CAN AVR

HOW TO START

Across this chapter we presume that you have a **GMM TST 2** or a **GMB HR84** where to install **CAN AVR**.
For further information please refer the specific manual of **GMB HR84 + CAN AVR**.

A) SERIAL CONNECTION BETWEEN CAN AVR AND PC

A1) To make the serial connection between **CAN AVR** and a PC, the structure described on figure 15 should be built.
The program delivered to the customer in the Mini Module is its test program, which communicates through USART0. Please remark that the test program is not designed to work outside grifo® specific test structures, so we strongly recommend this: **do not connect any other signal** exept the ones of USART0 in this phase.

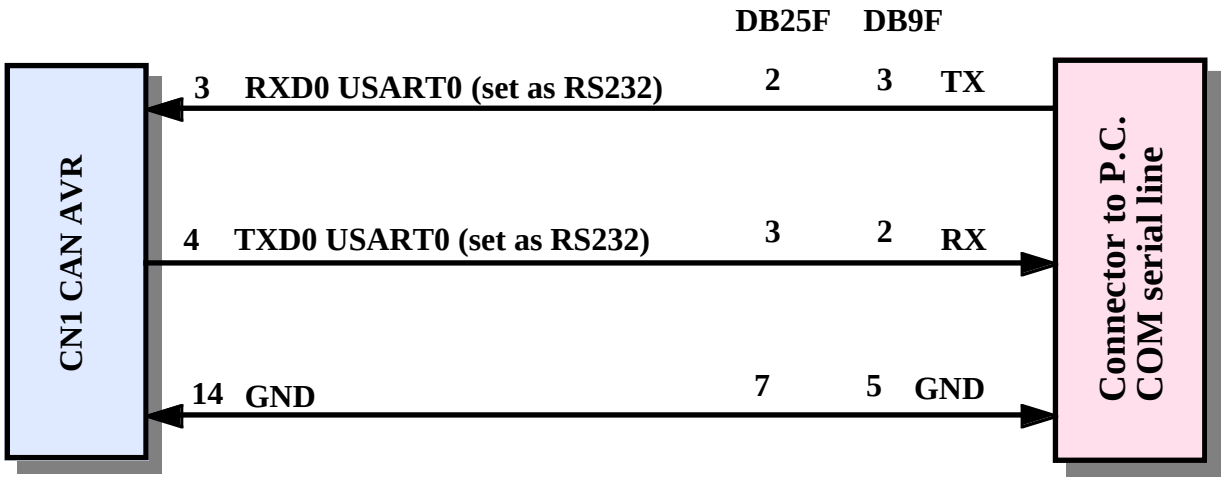


FIGURE 15: RS 232 SERIAL CONNECTION BETWEEN A CAN AVR AND A PC

A2) Keep ready for running a terminal emulator on PC, configure it to use the serial port where Mini Module is connected with 19200 baud, 8 data bits, 1 stop bit, no parity. If you are using BASCOM AVR, you may simply open the terminal emulator in its IDE.

A3) Supply **GMM TST 2** or **GMB HR168**. Starting screen of **CAN AVR** test program should appear in the terminal. Otherwise, check the connection cable and power supply.

BOARD TYPE	GET	ASM	Ladder	Abaco® Link BUS	BASIC CBZ80	BASIC BASCOM 8051	BASIC BASCOM AVR	PIC BASIC	BASIC VARI	MCS® Basic 52	C	PASCAL	CPU / BLOCK TYPE
MISCELL.	-	-	-	-	-	●	-	-	-	-	-	-	-
CAN GM0	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc03 - 8051 Code
CAN GM1	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc01 - 8051 Code
CAN GM2	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc02 - 8051 Code
GMM 5115	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C5115 - 8051 Code
GMM 876	●	-	-	-	-	-	-	●	-	-	-	-	Microchip PIC16F876A - PIC 14 Code
GMM 932	●	-	-	-	-	●	-	-	-	-	-	-	PHILIPS P89LPC932 - 8051 Code
GMM AC2	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51AC2 - 8051 Code
GMM AM08	●	-	-	-	-	-	●	-	-	-	-	-	Atmel ATMega08 - AVR Code
GMM AM32	●	-	-	-	-	-	●	-	-	-	-	-	Atmel ATMega32 - AVR Code
GMB HR84	●	-	-	-	-	●	●	●	-	-	●	-	Mini Block 8 input opto 4 output relè
GMB HR168	●	-	-	-	-	●	-	-	-	-	●	-	Mini Block 16 input opto 8 output relè

FIGURE 16: EXAMPLES TABLE

B) FLASH REPROGRAMMING

Mini Module programming is performed using th Boot Loader preprogrammed in the Flash of Mini Module itself.
Such program allows to create a communication between Mini Module and the PC, and to use it to send a .hex file to code memory and/or EEPROM memory area of microcontroller.
Boot Loader does not allow to program configuration bits and security fuses of microcontroller, this can be done usigne ISP programming (please refer to appendix B of this manual).
The portion of Flash memory taken by the Boot Loader is 4 KBytes, this means that the last block of 2 KWord at the end of the memory is reserved to Boot Loader and is not available for user application any more.
Boot Loader communicates through PC serial port installing a specific utility (that can be downloaded for free from our website www.grifo.com or can be found in our CD) called AVRBootloaderGrifo.exe. This utility allows to select the serial port to use end to select the files to program memories of Mini Module.
For some applications, use of an ISP programmer may be the only chance. Please refer to appendix B for more information about ISP programming.
Combo box called "Com Port" allows to select the serial port to use for communication with Mini Module.
Checking the ckeck box "Application Code" the file whose name is written in the text box on the right will be stored in Flash of Mini Module. To choose the file press the "Browse" button.
Chek box "EEPROM Code" performs the same operation described above, but writes to EEPROM.

There is also the possibility to use AVRBootloaderGrifo from DOS window, commands prompt, a command line or as an external tool of an IDE.
Option for command line are:

/com1 , /com2 , ... , /com9	Number of serial port used for communication. Serial ports from COM1 to COM9 are supported
/f <FLASHProgram>	Indicates the complete pathname of the .HEX file to write in microcontroller FLASH memory
/e <EEPROMProgram>	Indicates the complete pathname of the .HEX file to write in microcontroller EEPROM memory

For example:
AVRBootloaderGrifo.exe /com2 /f C:\Projects\MotorControl\Main.hex
opens a connection on serial port COM2 to the Boot Loader and sends the file Main.hex located in folder C:\Projects\MotorControl\.
To integrate AVRBootloaderGrifo.exe in a IDE, for example the one of BASCOM AVR (described at point C) the user must:

1) Open the window of menu Options | Programmer
2) In the text box Programmer choose "External Programmer"
3) Click the tab Other
4) In text box Program insert the complete pathname of AVRBootloaderGrifo.exe pressing Browse
5) In text box Parameters insert command line parameter for AVRBootloaderGrifo
Please remaind that the string {file} in the above mentioned text box is automatically replaces by BASCOM AVR with the name of .hex file just generated.
Please refer to BASCOM AVR documentation for further information.

B1) Find on CD grifo® and save to a comfortable position on your hard drive the demo program"prCANAVRuk.hex". It can be found starting from main page following the path: English | Examples tables | Mini Modules and Mini Block examples | CAN AVR (please refer to figure 16). You may want to remove the read-only attribute.

B2) Connect the Mini Module to PC serial port makin the connection described at point A. Close the terminal emulator.

B3) Indicate in combo box "Com Port" the serial port connected at the previous point.

B4) Check the check box "Application Code" and press the button "Browse" on the right, then select the file previously saved at point B1.

B5) Uncheck the check box "EEPROM Code", if checked.

B6) Press the button "Synch to Bootloader..." or the key combination Alt+S on the PC, then reset the Mini Module or turn off and then on its supply. The file is dowloaded into the Mini Module. If this does not happen, and the program should indicate a "No repsonse from target bootloader", try to repeat the operation decreasing the time between pressure of button on the PC and reset of Mini Module. If the problem persists, check cable and connection.

B7) When operation is completed the program reports its status. In case of problems, check cable and connection.

B8) Start the terminal emulator configured like in point A2 and verify that the application program just downloaded is executed in internal Flash.

FIGURE 17: BOOT LOADER UTILITY

C) GENERATING DEMO EXECUTABLE CODE

C1) Install on the hard disk of the development P.C. the software environment selected to develop the application program. As described in the chapter SOFTWARE DESCRIPTION there are many different software tools that satisfy any customers requirements but here we remind only the most diffused as the BASCOM AVR, ICC AVR, etc.

C2) On **grifo®** CD in addition to file with the executable code of the demo program, described at point B2, there are also the source files of the same. These have an extension that identifies the used software development tools (for example prCANAVRuk.bas for BASCOM AVR or prCANAVRuk.c for ICC AVR) and they are properly organized inside demo programs tables available on CD, together with possible definition file (for example: prCANAVRuk.prj for ICC AVR). Once these files have been located they must be copied in a comfortable folder on the hard disk of development PC.

C3) Compile the source file by using the selected software tools: the file prCANAVRuk.hex must be obtained equal to those available on **grifo®** CD and already used at points B. This operation is very different according to the programming environment selected, so here follow the details:

C3 Bascom AVR) *Ricompilation using BASCOM AVR.*

C3 Bascom AVR a) In BASCOM IDE, load the program source with menu File | Open:

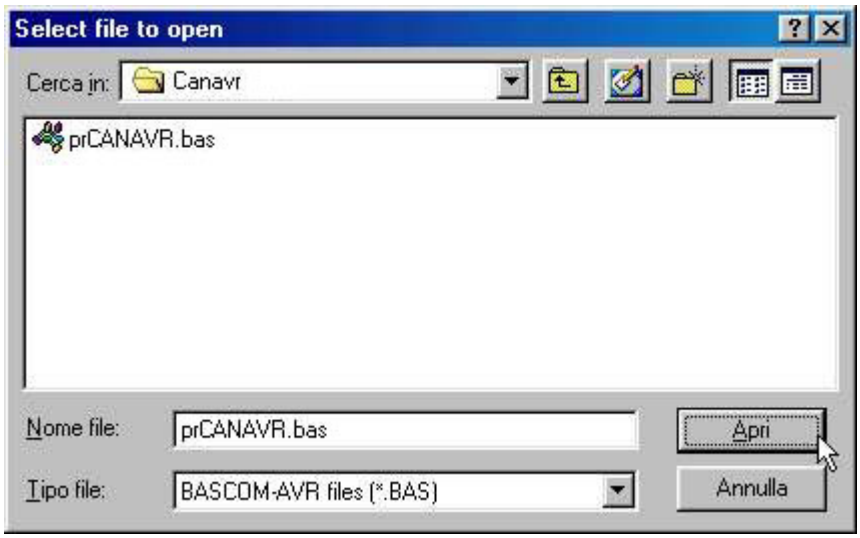


FIGURE 18: LOADING A SOURCE FILE WITH BASCOM AVR

C3 Bascom AVR b) From menu Options | Compiler | Chip set the value 64 for HW Stack, 32 for Soft Stack, 64 for Framesize, as suggesterd also in the source code, and press OK. Such values must be considered minimal and must be increased if required:

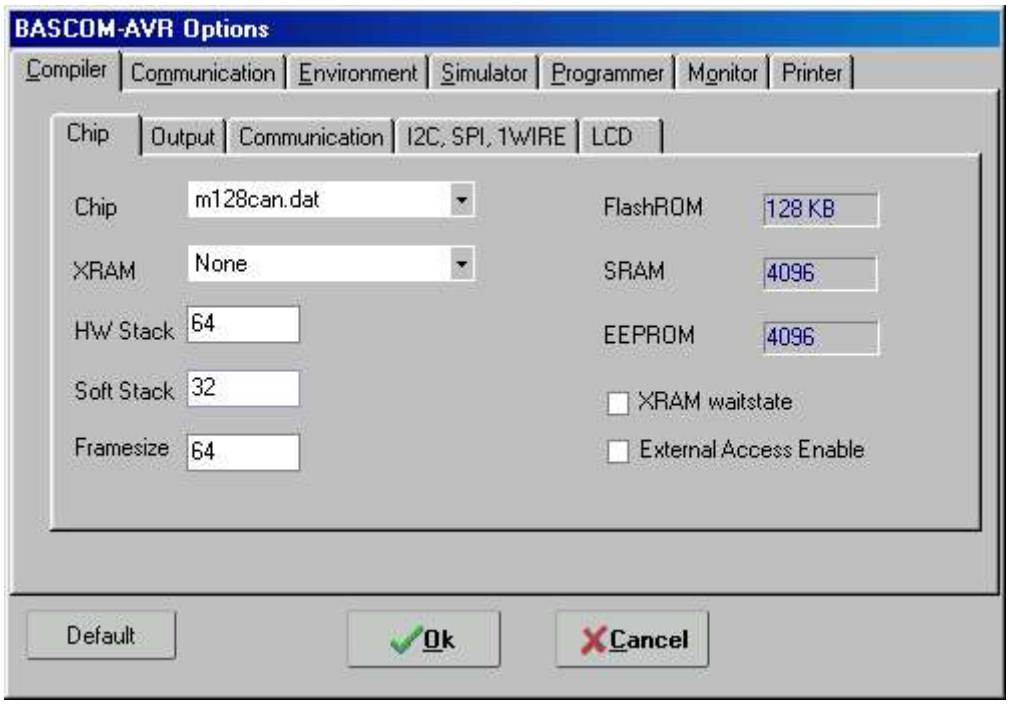


FIGURE 19: CONFIGURATION OF COMPILER BASCOM AVR

C3 Bascom AVR c) Compile the source file by pressing the button with the icon of an integrated circuit.

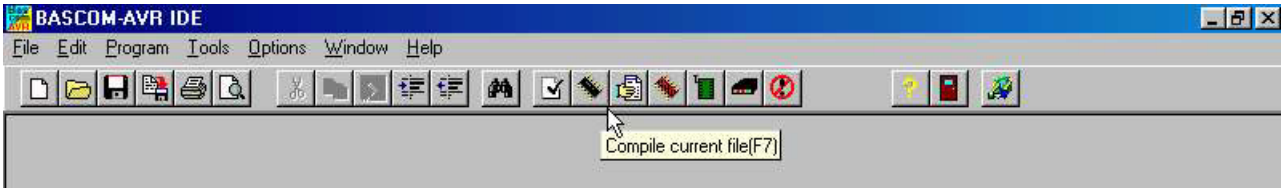


FIGURE 20: COMPILATION WITH BASCOM AVR

C3 ICC AVR) *Ricompilation with ICC AVR.*

C3 ICC AVR a) In standard editor, load the project file prCANAVRuk.prj using the menu Project | Open...:

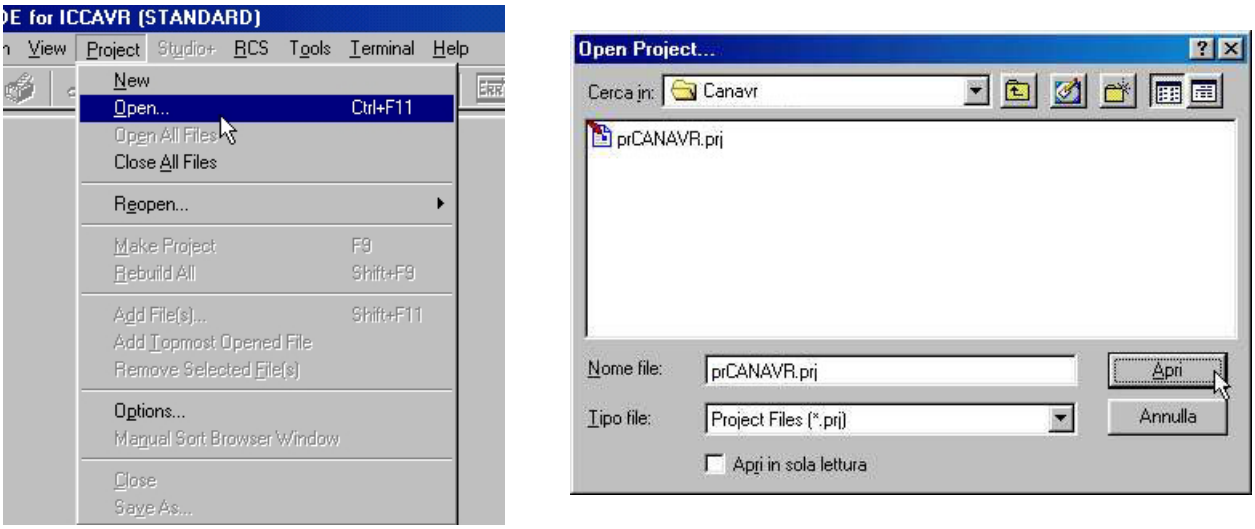


FIGURE 21: LOADING PROJECT FILE WITH ICC AVR

C3 ICC AVR b) Compile the project using the menu Project | Make project:

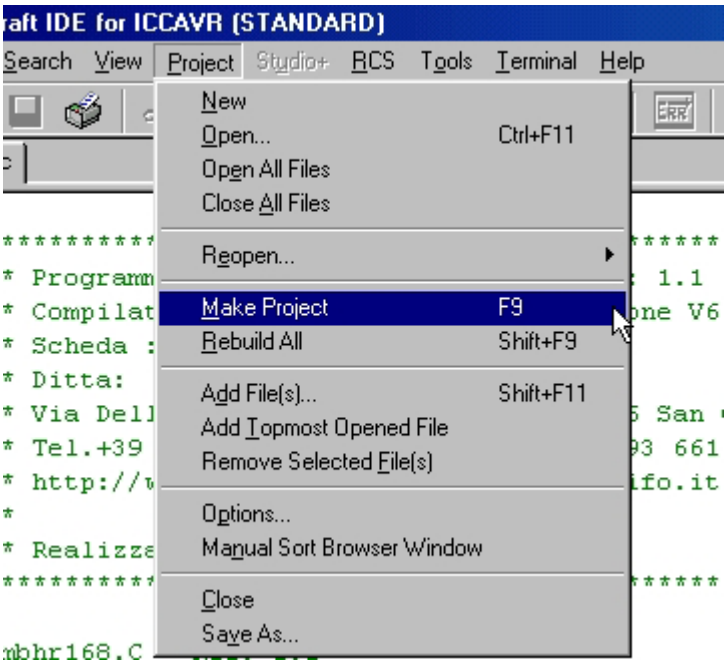


FIGURE 22: COMPILATION WITH ICC AVR

C4) Program the compiled file into FLASH memory of **CAN AVR** repeting the steps of point B.

D) *FINAL APPLICATION*

D1) Close the Boot Loader PC utility.

When during execution of the steps above described a problem or a malfunction is found, we suggest to read and repeat again all the steps carefully and if malfunction persists please contact directly **grifo®** technician.

Instead when execution of all the steps above described is right, the user has realized his first application program that coincides with demo of **CAN AVR**.
At this point it is possible to modify the source of the demo/s program according to application requirements and test the obtained program with the steps above listed (successive to B and C) in cyclic mode, until the developed application program is completely well running.
When this focus is reached the developmnet PC can be eliminated.
Remember to reconfigure USART0 of Mini Module, if required.

SOFTWARE DESCRIPTION

A wide selection of software development tools can be obtained, allowing use of the module as a system for its own development, both in assembler and in other high level languages; in this way the user can easily develop all the requested application programs in a very short time.
Generally all software packages available for the mounted microprocessor, or for the AVR family, can be used.
Software packages purchased from **grifo®** are always provided with example programs that show how to use each section of the board and a complete use documentation.
Remarkable are:

BASCOM AVR
It is a powerfull new integrated development environment for AVR microcontroller. The toolset incorporates an editor, optimising BASIC compiler, assembler and HEX creator. The BASIC compiler produces very tight AVR machine code by virtue of the fact it translates the BASIC source into actually run timeassembly code wich is optimised to run as fast as possible. The target AVR microcontroller therefore runs true assembly code rather than tokenised code wich is found in many other BASIC compilers. It is also provided with integrated simulator for source level debugging and optional external libraries to drive or simulate several external devices (likebadge readers, PS/2 keyboards, graphic and alphanumeric displays, etc.).

ICC AVR PRO
Cross compiler for C source program. It is a powerfull software tool that includes editor, ANSI C compiler, assembler, linker, library management program and project manager included in an easy to use integrated development environment for Windows and other P.C. operating systems. Library sources, floating point, integration with AVR studio, on line help and ANSI terminal emulator for target communication are provided too.

DDS MICRO C AVR
Low cost ross compiler for C source program. It is a powerful software tool that includes editor, C compiler (integer), assembler, optimizer, source linker and library in one easy to use integrated development environment. There are also included the library sources and many utilities programs. The default IDE can be replaced by a new one named **Micro IDE**, that is more powerfull, for Windows operating system and provided of many utility functions.

AVR Studio
It is a development tool for AVR family of microcontroller that fully control execution of program on AVR in circuit emulator or on the built in AVR instruction set simulator. AVR Studio supports source level execution of assembly and C programs generated by external compilers and assemblers. The tolls is based on a set of windows for source, watch, registers, memory, peripherals, message and processor that enable the user to have full control of the status of every element in the execution target. It also features an "application builder" to easy the generation of code to initialise all hardware peripherals (USART, SPI, Port, ADC, ect.) starting from a graphic interface.

There is also the remarkable possibility to drive the JTAG interface called "JTAG ICE" manufactured by Atmel.
A JTAG interface allows to enter the core of microcontroller to examine its status during execution of applcation program directly on the application hardware.
The user can insert both hardware and software breakpoints, and when execution is stopped the values contained in memory and internal registers can be examined.
JTAG interface also allows to reprogram the microcontroller memories.
Using JTAG interface several debugging problems are solved, increasing the possibility to eliminate bugs and drastically reducing the time required to obtain the final application completely debugged.

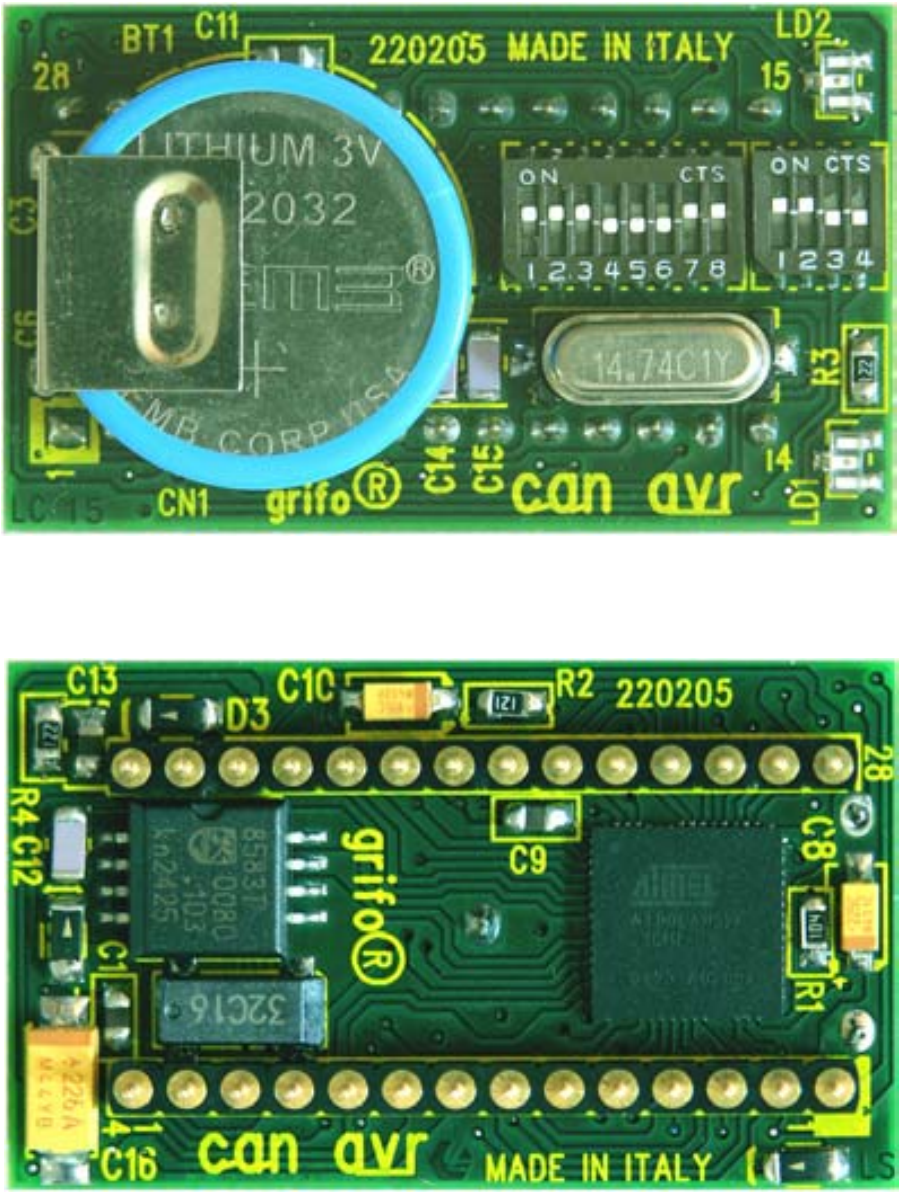


FIGURE 23: TOP VIEW AND BOTTOM VIEW OF CAN AVR

PERIPHERAL DEVICES SOFTWARE DESCRIPTION

Below there is a specific description of the software managements of the on board peripheral devices. Whenever the reported documentation is not sufficient, please search a more detailed description of the devices in manufacturing company data sheets. Furthermore in this chapter the microprocontroller internal peripheral devices are not described so if their programmation is necessary, please refer to appendix A of this manual.
In the following paragraphs the **D7÷D0** and **.0÷7** indications denote the eight bits of the combination involved in I/O operations.

ACTIVITY LEDS

The **CAN AVR** allows software management of activity or status LEDs LD1 and LD2, through an I/O line of the microcontroller, with the following corrispondence:

PA0 = 0	->	**LD1 ON**
PA0 = 1	->	**LD1 OFF**
PA1 = 0	->	**LD2 ON**
PA1 = 1	->	**LD2 OFF**

It is important to remind that PA0 and PA1 are not connected to CN1.
The signals PA0 and PA1 are set high after reset or power on, so during these phases LEDs are OFF or disabled.

DIP SWITCH

CAN AVR allows to read by software the status of two dip switches, DSW1.7 e 8, through two I/O signals of microcontroller:

DSW1.7 ON	->	**PA2 is 0**
DSW1.7 OFF	->	**PA2 is 1**
DSW1.8 ON	->	**PA3 is 0**
DSW1.8 OFF	->	**PA3 is 1**

It is important to remind that PA2 and PA3 are not connected to CN1.

BACKED SRAM + SERIAL RTC

Mini Module **CAN AVR** is provided with a complete Real Time Clock capable to manage hours, minutes, seconds, day, month, year and weekday in complete autonomy.
This module can be backed up through the back up circuitery to warrant data validity in any working condition and is completely managed by software.
RTC section can also generate periodic interrupts at software programmable time intervals, to switch the CPU out of normal operations or to awaken it from low consumption modes.
For software management of serial SRAM + RTC backed module, please refer to specific manufacturer documentation. This manual reports no software information because management of this component is complex and requires a deep knowledge, anyway the user can use the demo programs supplied with the card. The board control logic allows to realize a serial communication with I^2C bus standard protocol, through two I/O microprocessor pins. The only necessary information is the electric connection:

PD1 , INT1 , SDA (input/output)	->	DATA line (SDA)
PD0 , INT0 , SCL (input/output)	->	CLOCK line (SCL)

Please remark that **A0** of this component's slave address is bound to logic 0.
This means that RTC takes permanently the slave address **A0**, which is not available for user application.
The above mentioned module is capable to generate a programmable periodic signal, which is connected on-board to a 4.7 kΩ pull-up, and is available on pin 11 of Mini Module connector CN1, and is connected to signal PD4 of microcontroller.

CPU INTERNAL PERIPHERALS

Registers description and purpose for all internal peripherals (Analog COMPARATORS, A/D CONVERTER, Timer Counters, USART0, CAN, I^2C BUS, SPI, OCM, etc.) is availabe in the proper data sheet and user manual of the manufacturer.
Please refer to chapter BIBLIOGRAPHY and to appendix A of this manual to easily locate such documentation.

BIBLIOGRAPHY

In this chapter there is a complete list of technical books and notes, where the user can find all the necessary documentations on the components mounted on **CAN AVR** Mini Module.

Technical paper MAXIM: *True RS 232 Transceivers*

Manual PHILIPS: *I2C-bus compatible ICs*

Technical paper ATMEL: *Data Sheet Atmel AT90CAN128*

The described manual can be requested directly to manufacturer or local dealers. Alternatively this information and/or upgrades can be found in specific internet web pages, of the listed companies.

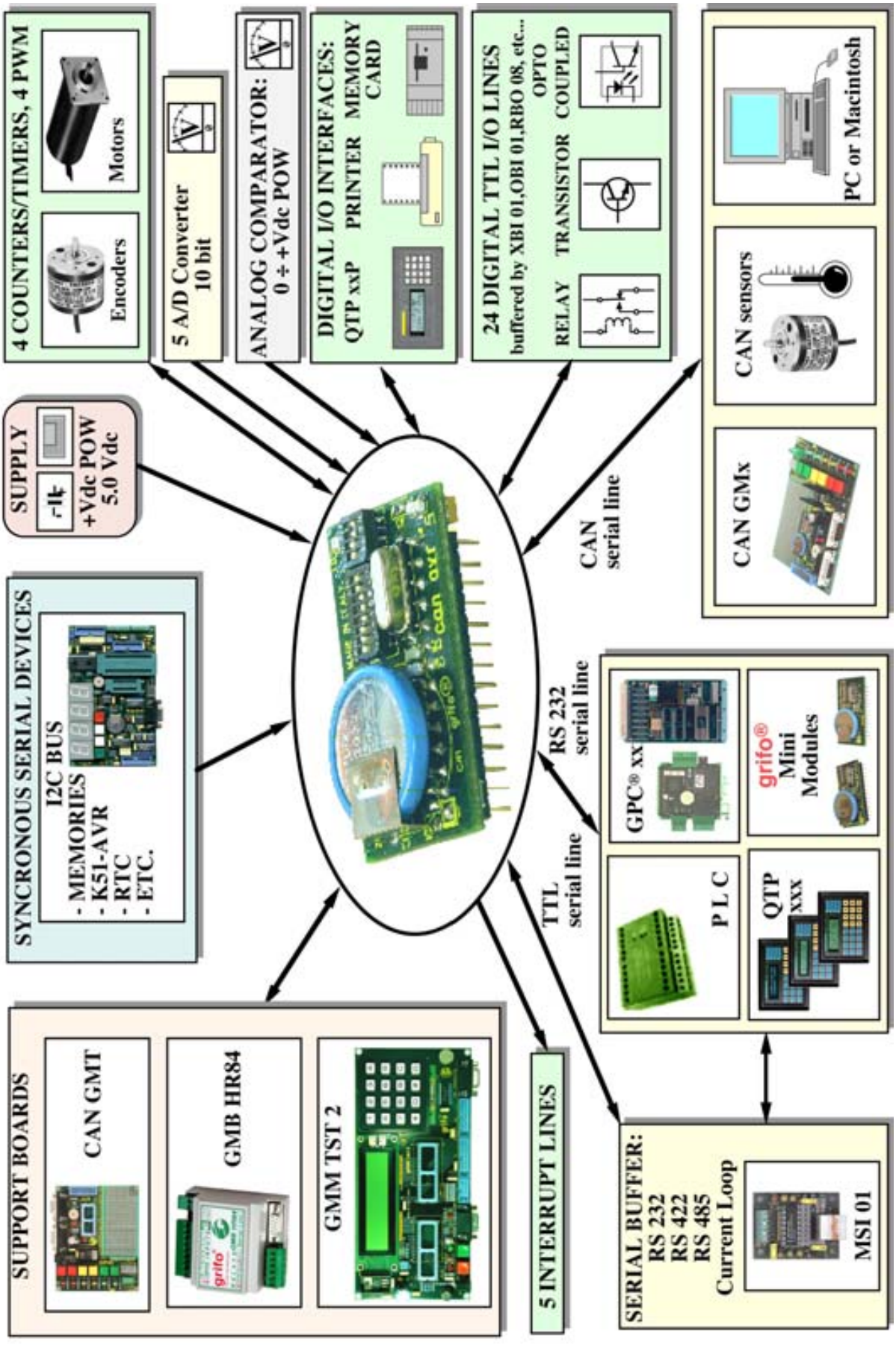

4 COUNTERS/TIMERS, 4 PWM
Motors
Encoders
5 A/D Converter 10 bit
ANALOG COMPARATOR: 0 ÷ +Vdc POW
DIGITAL I/O INTERFACES: QTP xxP PRINTER MEMORY CARD
24 DIGITAL TTL I/O LINES buffered by XBI 01,OBI 01,RBO 08, etc... OPTO RELAY TRANSISTOR COUPLED
SUPPLY +Vdc POW 5.0 Vdc
SYNCRONOUS SERIAL DEVICES I2C BUS - MEMORIES - K51-AVR - RTC - ETC.
SUPPORT BOARDS
CAN GMT
GMB HR84
GMM TST 2
5 INTERRUPT LINES
SERIAL BUFFER: RS 232 RS 422 RS 485 Current Loop
MSI 01
TTL serial line
RS 232 serial line
CAN serial line
GPC® xx
grifo® Mini Modules
P L C
QTP xxx
CAN sensors
CAN GMx
PC or Macintosh

FIGURE 24: AVAILABLE CONNECTIONS DIAGRAM

APPENDIX A: DATA SHEETS

grifo® provides a completely free technical documentation service to make available data sheets of on board components, through its web site. In this chapter the user found the complete and ready to use links and URLs to these information, together with the first pages of the same documents. To use our technical documentation service just connect to our site www.grifo.com and click its icon.

AT90CAN128

Link: Home | Technical documentation Service | ATMEL | Data-Sheet AT90CAN128
URL: http://www.grifo.com/PRESS/DOC/Atmel/AT90CAN128.pdf

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 133 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers + Peripheral Control Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Non volatile Program and Data Memories
  - 128K Bytes of In-System Reprogrammable Flash
    Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    Selectable Boot Size: 1K Bytes, 2K Bytes, 4K Bytes or 8K Bytes
    In-System Programming by On-Chip Boot Program (CAN, UART)
    True Read-While-Write Operation
  - 4K Bytes EEPROM (Endurance: 100,000 Write/Erase Cycles)
  - 4K Bytes Internal SRAM
  - Up to 64K Bytes Optional External Memory Space
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Programming Flash (Hardware ISP), EEPROM, Lock & Fuse Bits
  - Extensive On-chip Debug Support
- CAN Controller 2.0A & 2.0B
  - 15 Full Message Objects with Separate Identifier Tags and Masks
  - Transmit, Receive, Automatic Reply and Frame Buffer Receive Modes
  - 1Mbits/s Maximum Transfer Rate at 8 MHz
  - Time stamping, TTC & Listening Mode (Spying or Autobaud)
- Peripheral Features
  - Programmable Watchdog Timer with On-chip Oscillator
  - 8-bit Synchronous Timer/Counter-0
    10-bit Prescaler
    External Event Counter
    Output Compare or 8-bit PWM Output
  - 8-bit Asynchronous Timer/Counter-2
    10-bit Prescaler
    External Event Counter
    Output Compare or 8-Bit PWM Output
    32Khz Oscillator for RTC Operation
  - Dual 16-bit Synchronous Timer/Counters-1 & 3
    10-bit Prescaler
    Input Capture with Noise Canceler
    External Event Counter
    3-Output Compare or 16-Bit PWM Output
    Output Compare Modulation
  - 8-channel, 10-bit SAR ADC
    8 Single-ended channels
    7 Differential Channels
    2 Differential Channels With Programmable Gain at 1x, 10x, or 200x
  - On-chip Analog Comparator
  - Byte-oriented Two-wire Serial Interface
  - Dual Programmable Serial USART
  - Master/Slave SPI Serial Interface
    Programming Flash (Hardware ISP)
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - 8 External Interrupt Sources
  - 5 Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down & Standby
  - Software Selectable Clock Frequency
  - Global Pull-up Disable
- I/O and Packages
  - 53 Programmable I/O Lines
  - 64-lead TQFP and 64-lead QFN
- Operating Voltages
  - 2.7 - 5.5V
- Operating temperature
  - Industrial (-40°C to +85°C)
- Maximum Frequency
  - 8 MHz at 2.7V - Industrial range
  - 16 MHz at 4.5V - Industrial range

ATMEL®

8-bit AVR® Microcontroller with 128K Bytes of ISP Flash and CAN Controller

AT90CAN128

Rev. 4250F–CAN–04/05

ATMEL

1

FEATURES

- ^{2}C-bus interface operating supply voltage: 2.5 V to 6 V
- :lock operating supply voltage (0 to +70 °C):
  .0 V to 6.0 V
- 40 × 8-bit low-voltage RAM
-)ata retention voltage: 1.0 V to 6 V
-)perating current (at f_{SCL} = 0 Hz): max. 50 μA
- :lock function with four year calendar
- Iniversal timer with alarm and overflow indication
- 4 or 12 hour format
- 2.768 kHz or 50 Hz time base
- ;erial input/output bus (I^2C)
- ,utomatic word address incrementing
- 'rogrammable alarm, timer and interrupt function
- ;lave address:
  - READ: A1 or A3
  - WRITE: A0 or A2.

2 GENERAL DESCRIPTION

The PCF8583 is a clock/calendar circuit based on a 2048-bit static CMOS RAM organized as 256 words by 8 bits. Addresses and data are transferred serially via the two-line bidirectional I^2C-bus. The built-in word address register is incremented automatically after each written or read data byte. Address pin A0 is used for programming the hardware address, allowing the connection of two devices to the bus without additional hardware.

The built-in 32.768 kHz oscillator circuit and the first 8 bytes of the RAM are used for the clock/calendar and counter functions. The next 8 bytes may be programmed as alarm registers or used as free RAM space.
The remaining 240 bytes are free RAM locations.

QUICK REFERENCE DATA

;YMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNIT
‹D	supply voltage operating mode	I^2C-bus active	2.5	–	6.0	V
		I^2C-bus inactive	1.0	–	6.0	V
›	supply current operating mode	f_{SCL} = 100 kHz	–	–	200	μA
›O	supply current clock mode	f_{SCL} = 0 Hz; V_{DD} = 5 V	–	10	50	μA
		f_{SCL} = 0 Hz; V_{DD} = 1 V	–	2	10	μA
‹mb	operating ambient temperature range		−40	–	+85	°C
‹g	storage temperature range		−65	–	+150	°C

ORDERING INFORMATION

TYPE IUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
:F8583P	DIP8	plastic dual in-line package; 8 leads (300 mil)	SOT97-1
:F8583T	SO8	plastic small outline package; 8 leads; body width 7.5 mm	SOT176-1

lips Semiconductors Product specification

Clock/calendar with 240 × 8-bit RAM PCF8583

BLOCK DIAGRAM

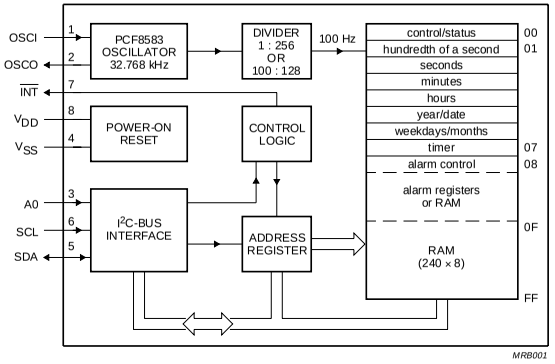


Fig.1 Block diagram.

PINNING

/MBOL	PIN	DESCRIPTION
;CI	1	oscillator input, 50 Hz or event-pulse input
;CO	2	oscillator output
	3	address input
;S	4	negative supply
)A	5	serial data line
;L	6	serial clock line
‾T	7	open drain interrupt output (active LOW)
‹D	8	positive supply

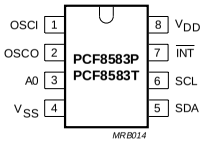


Fig.2 Pinning diagram.

Page A-2
CAN AVR Rel. 5.00
PCF 8583
Link: Home | Technical documentation Service | I²C BUS | Data-Sheet PCF8583
URL: http://www.grifo.com/PRESS/DOC/PHILIPS/PCF8583.pdf
grifo®
ITALIAN TECHNOLOGY

APPENDIX B: ISP PROGRAMMING WITH GMM TST 2

In this appendix the user can find specific instructions to program Mini Module **CAN AVR** installed on a **GMM TST 2** using an external ISP programmer.
This method to program Mini Module on board Flash and EEPROM can be considered and alternative to the one explained in section "B" of chapater "How to start" of manual.
Content of other sections in above mentioned chapter are still valid anyway.
ISP programming allows also to change the value of configuration bits and security fuses, in addition to the programming of whole Flash and EEPROM content.
The Boot Loader does not allow to program the whole content of Flash, because the Boot Loader itself resides in a Flash area which becomes protected.
Also, configuration bits, that enable the Boot Loader itself, and security fuses are out of the range programmable by the Boot Loader and must be programmed in ISP mode.
If you think it is more convenient to use a Boot Loader on RS 232 serial line, refer to above mentioned chapter.
For more information on how to connect a PC serial port to **GMM TST 2** (or even to Mini Module as stand-alone) to test the program, refer to section "A" of chapter "How to start" of manual.
For further information on how to develop and debug the firmware, refer to section "C" of chapter "How to start" of manual.
Programmers that can be used with **GMM TST 2** are:

- **grifo® MP-AVR 51+** ; **grifo® UEP 49**
- Equinox EPSILON5
- ATMEL AVR ISP

A) *COMMON OPERATIONS FOR ALL PROGRAMMERS*

A1) Programming of Mini Module on a **GMM TST 2** board is performed through a specific connector of the card and, if required, an interface adapter between the programmer and the connector itself.
The two **grifo®** programmers that can be interfaced with **GMM TST 2**, that is **UEP 49** and **MP-AVR 51+**, use a specific interface for each one of them, whose diagram is shown in figures B-1 and B-2. On the left side of the figures there is the list of **GMM TST 2** CN7 connector's ways that must be connected to corresponding pins on programmer's connector indicated by the arrow.
Programmers AVR ISP and Equinox EPSILON5 do not require any specific interface.

A2) Interfacement between Mini Module **GMM AM 128** and on of the 40 ways socket connectors of **GMM TST 2** is performed through a specific ISP programming socket. To perform ISP programming of **GMM AM128** on socket Z1 or Z2 of a **GMM TST 2**, ISP programming socket must be installed between **GMM TST 2** and Mini Module. ISP programming socket must connect pins 14 and 15 of **GMM TST 2** respectively to pins 9 and 10 of Mini Module, and isolate pins 9 and 10 of **GMM TST 2**, as indicated in figure B-4.

A3) **GMM TST 2** jumpers configuration and Mini Module Dip Switch position is the one reported on page 24 of manual and repeated here for comfort:

Configuration ***GMM AM128***			*Configuration* ***GMM TST 2***		
DSW1.1	=	OFF	J1	=	2-3
DSW1.2	=	OFF	J2	=	2-3
DSW1.3	=	OFF	J3	=	not connected
DSW1.4	=	ON	J4	=	not connected
DSW1.5	=	ON	J5	=	not connected
DSW1.6	=	OFF	J6	=	not connected
DSW1.7	=	OFF	J7	=	not connected
DSW1.8	=	OFF			
DSW2.1	=	OFF			
DSW2.2	=	OFF			
DSW2.3	=	OFF			
DSW2.4	=	OFF			

A4) Locate on **grifo®** CD and save to a comfortable position on the hard disk of the PC the file called "prCANAVRuk.hex" following the path: English | Example Tables | Mini Module and Mini Block examples | CAN AVR (please refer to figure 16 of manual).

Next sections described a detailed configuration specific for each programmer.

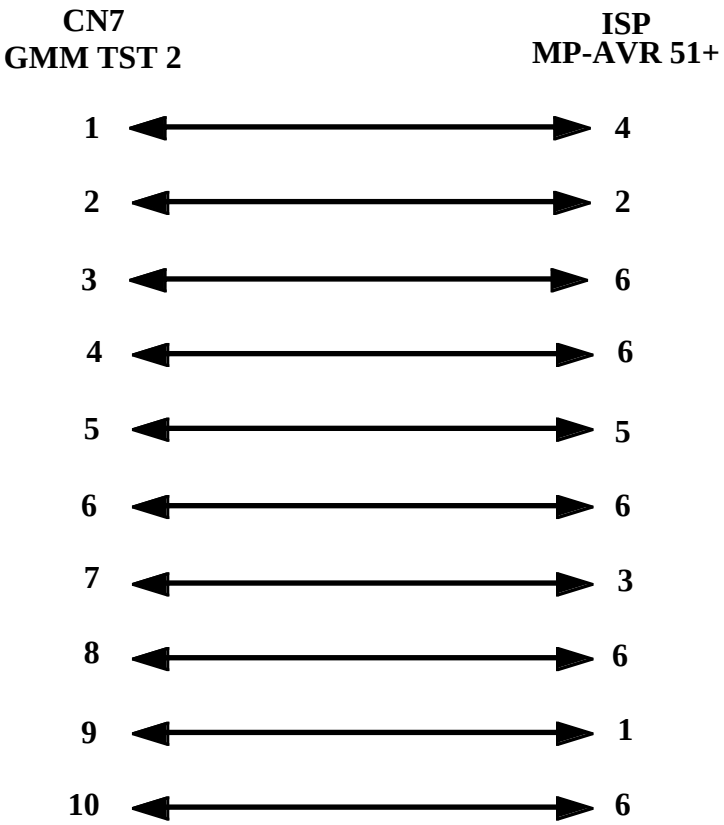


FIGURE B-1: DIAGMAM OF INTERFACE BETWEEN GMM TST 2 AND MP-AVR 51+

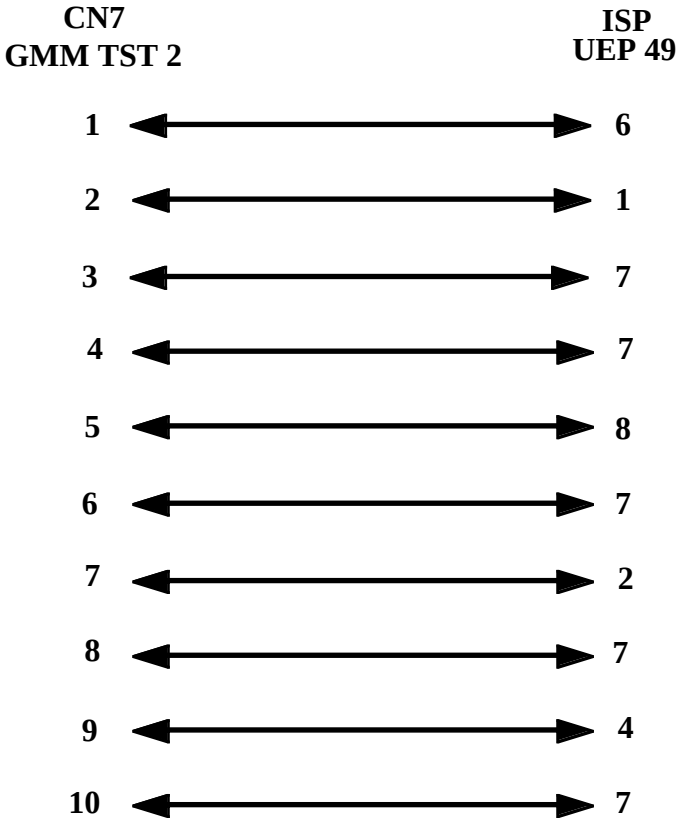


FIGURE B-2: DIAGMAM OF INTERFACE BETWEEN GMM TST 2 AND UEP 49

B) PROGRAMMING USING GRIFO® MP-AVR 51+ ; GRIFO® UEP 49

B1) Instructions for installing, connecting to PC and using the programmer are delivered with the programmer itself. The control program for all **grifo®** programmers is called PG4UW and will be indicated by that name from now on.

B2) Connect the programmer to **GMM TST 2** using the interface described at point A2.

B3) Insert module in socket Z1 or Z2 of **GMM TST 2** using the interface and the instructions described at point A2 and A3, then turn on power supply of **GMM TST 2** as described in the manual of **GMM TST 2**.

B4) Select AT90CAN128 ISP as the component to program in specific menu of PG4UW, as indicated in figure B-3.

B5) Load the file previously saved on point A4) by pressing the button "Load".

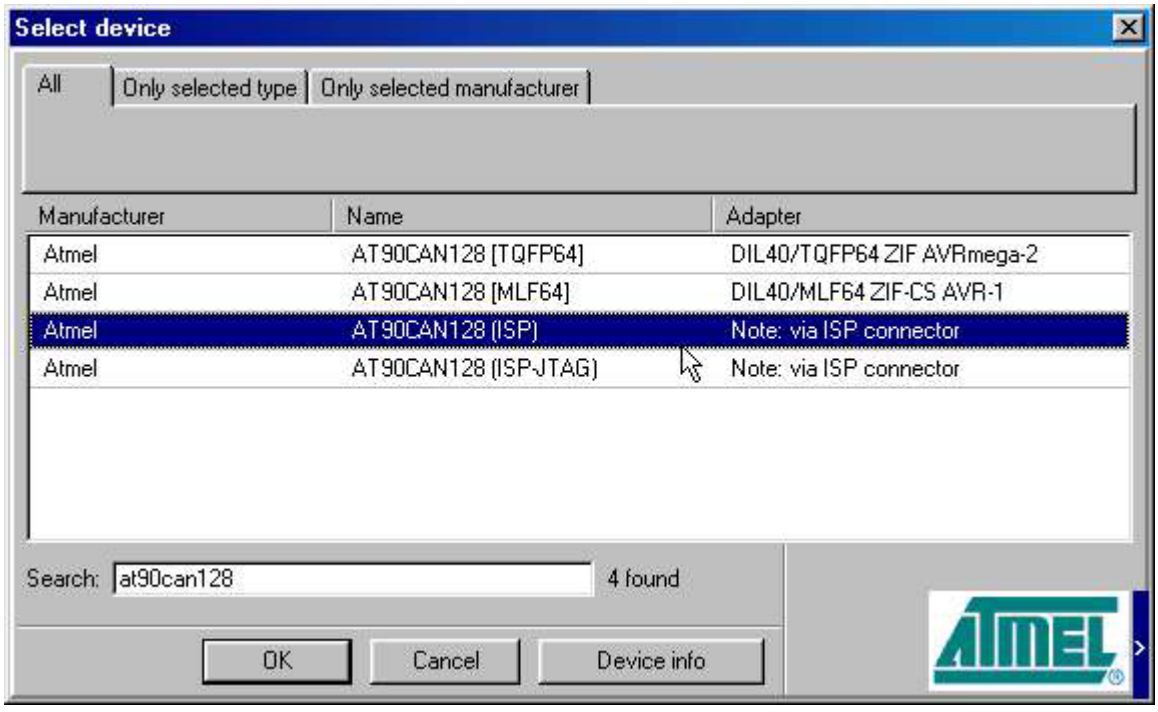


FIGURE B-3: COMPONENT SELECTION USING PG4UW

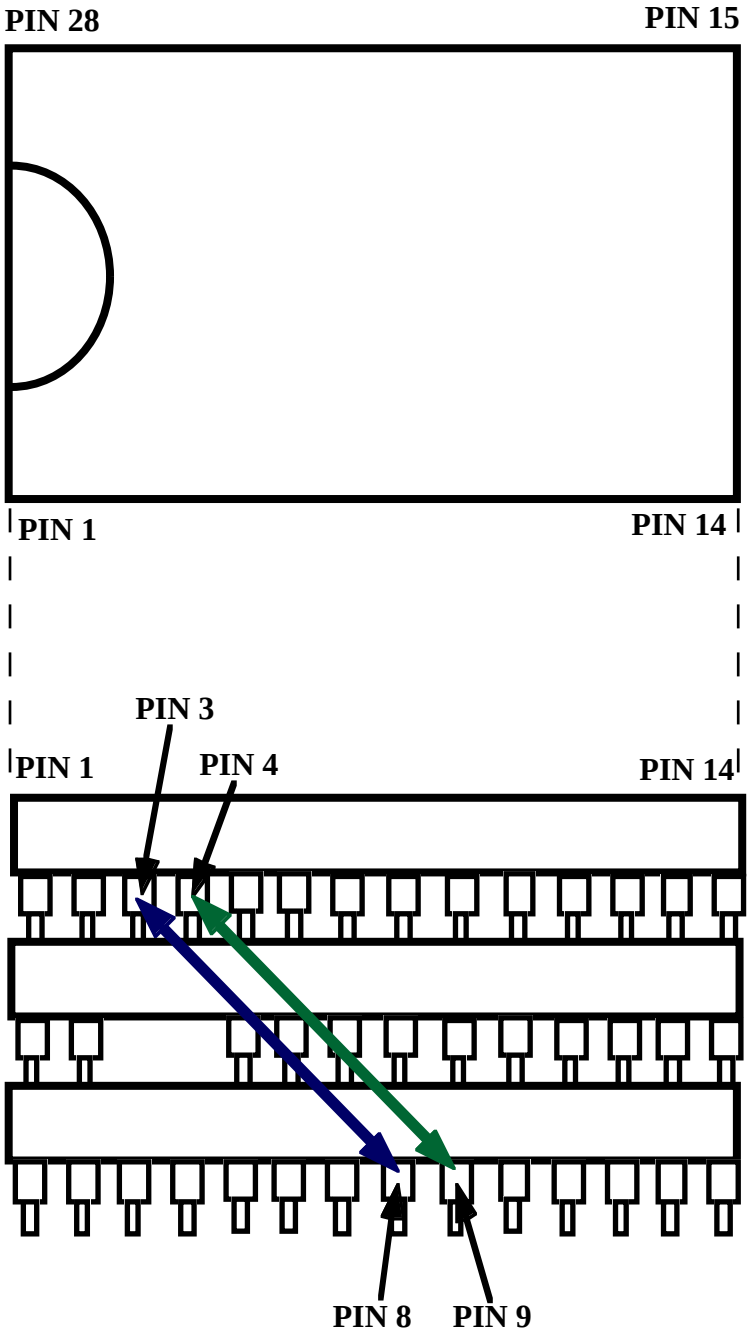


FIGURE B-4: ISP PROGRAMMING SOCKET FOR CAN AVR AND GMM TST 2

B6) Configure the component using the menu Device options | View/Edit options and security. The component must be configured as indicated in figure B-5, that is the configuration required to make the demo program prCANAVRuk.hex work. Quartz and start-up time options must be configured like in the figure. Other settings are not important. Of course, the user can perform any other configuration required by his/her application.

B7) Configure the programmer using the menu Device options | Operating options. Programmer must be configured as indicated in figure B-6.

B8) Start the programming operation, pressing the "Program" button.

B9) During the programming operation, a progress bar advances until the operation is completed. Eventual errors are promptly indicated.

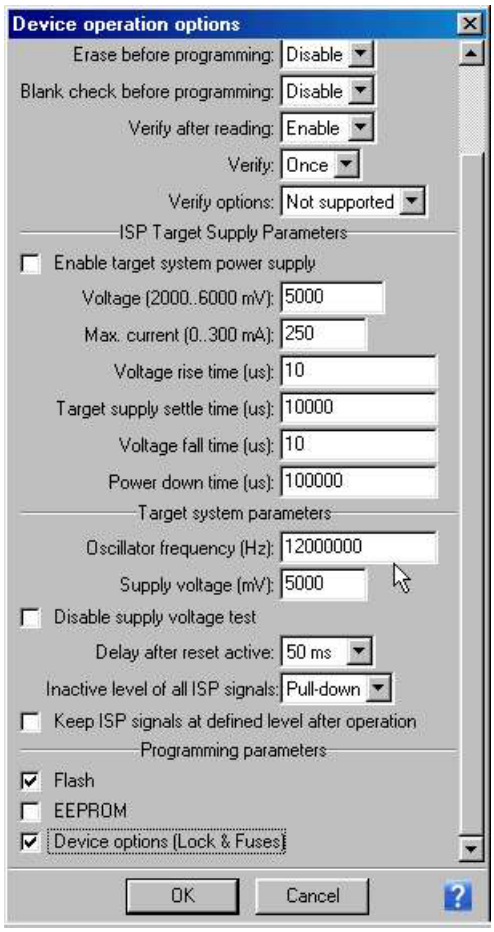

Device operation options
Erase before programming: Disable
Blank check before programming: Disable
Verify after reading: Enable
Verify: Once
Verify options: Not supported
ISP Target Supply Parameters
Enable target system power supply
Voltage (2000..6000 mV): 5000
Max. current (0..300 mA): 250
Voltage rise time (us): 10
Target supply settle time (us): 10000
Voltage fall time (us): 10
Power down time (us): 100000
Target system parameters
Oscillator frequency (Hz): 12000000
Supply voltage (mV): 5000
Disable supply voltage test
Delay after reset active: 50 ms
Inactive level of all ISP signals: Pull-down
Keep ISP signals at defined level after operation
Programming parameters
Flash
EEPROM
Device options (Lock & Fuses)
OK
Cancel

FIGURE B-5: PROGRAMMER CONFIGURATION USING PG4UW

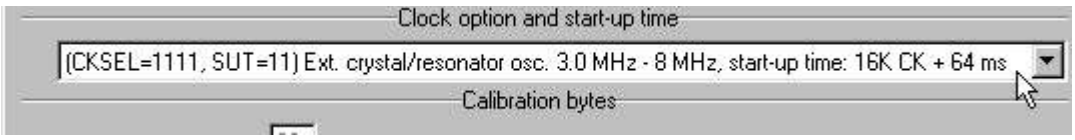

Clock option and start-up time
(CKSEL=1111, SUT=11) Ext. crystal/resonator osc. 3.0 MHz - 8 MHz, start-up time: 16K CK + 64 ms
Calibration bytes

FIGURE B-6: COMPONENT CONFIGURATION USING PG4UW

C) PROGRAMMING USING EQUINOX EPSILON5

C1) Programming of Mini Module on a **GMM TST 2** board is performed through a specific connector of the card. First of all, programmer Equinox EPSILON5 must be physically configured to connect to **GMM TST 2**. To perform such configuration:

- Connect the 10 ways flat cable to connector "J7-ATMEL10" of EPSILON5
- Connect jumper J9 of EPSILON5

for further information please refer to manual of EPSILON5.

C2) After configuring and closing the EPSILON5, its control program must be installed. This program's name is EQTools, and will be called this way from now on.For further information about installing the program and connecting EPSILON5 to the PC, please refer to manual of EPSILON5.

C3) Connect the programmer to connector CN7 of **GMM TST 2**.

C4) Insert module in socket Z1 or Z2 of **GMM TST 2** using the interface and the instructions described at points A2 and A3, then turn on power supply of **GMM TST 2** as described in the manual of **GMM TST 2**.

C5) To use Equinox EPSILON5 connected to the PC (instead of stand-alone), it is required to open a project file (extension EDS). This can be done creating a new one from stat screen of EQTools, using specific menus and buttons or loading and existing project file. For further information about project file management, please refer to manuale of EQTools.

C6) Who creates a new project, must be sure to perform the settings shown in figure B-7. Who opens an existing project must be sure that these settings have already been performed. These settings assure that:

- The project is programmed correctly by EPSLON5
- Target device is AT90CAN128
- File programmed on the target is prCANAVRuk.hex
- Target component configuration is correct

for further information about configuring a project, please refer to EQTools documentation.

C7) Who creates a new project, must also indicate to use it as "Test EDS" mode. To enable and perform Flash write operation the user must move back to Flash menu, put the check sign on "Edit Menu" checkbox and press button "Write".

C8) Window "Write Block to Flash" shows a summar of some current settings to verify them. If they are correct, pressing OK starts the memory writing procedure.

C9) Current status of programming is indicated by a progress bar, when programming completes a message indicates the operation final result.

C10) To perform configuration bits write operation it is required to move back to Fuses menu, and press button "Write" in frame "Target Fuses".

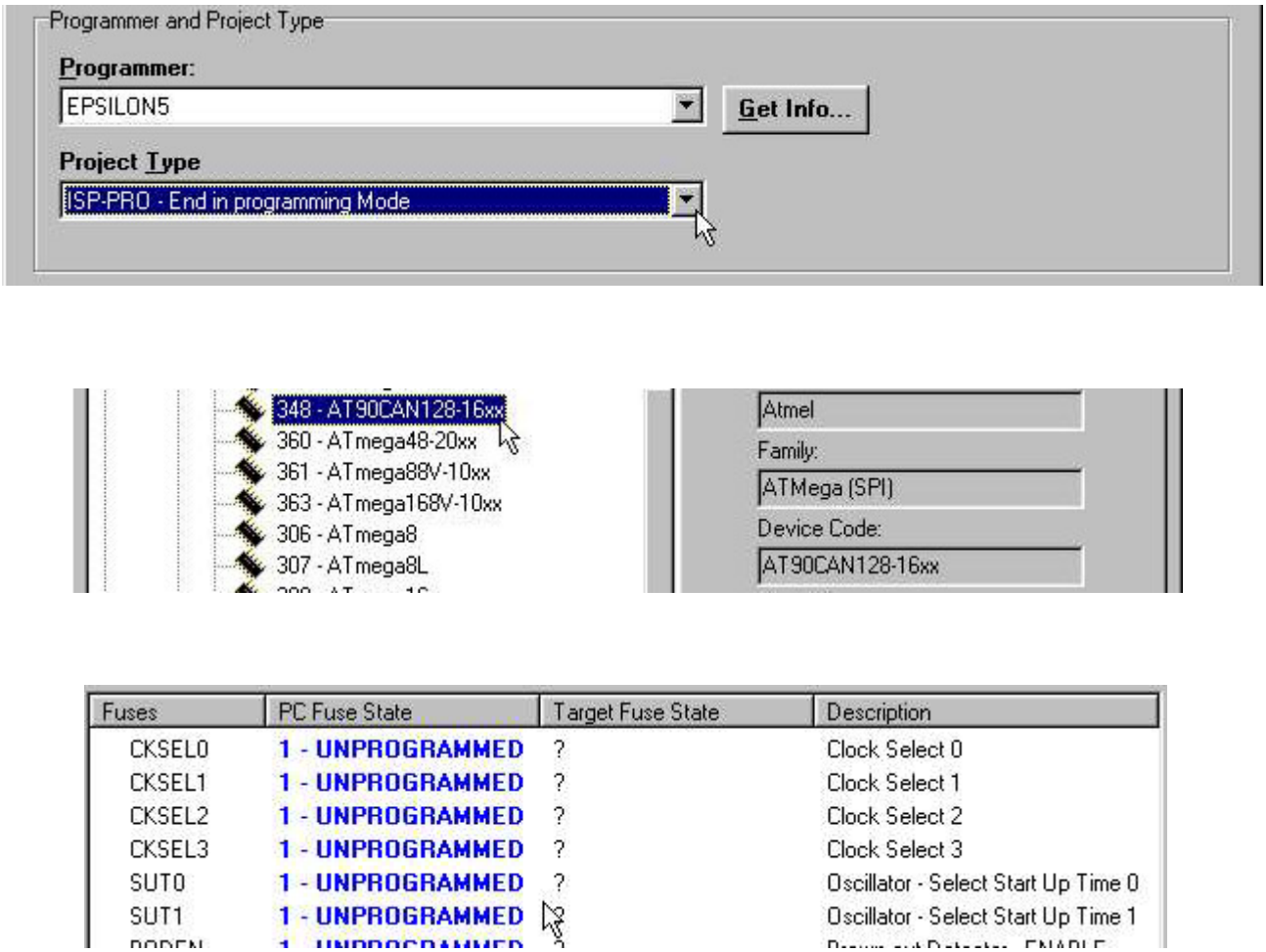


FIGURE B-7: PROJECT CONFIGURATION USING EQTOOLS

D) PROGRAMMING USING ATMEL AVR ISP

D1) Control program of **AVR ISP** is **AVR STUDIO**, version 4 or greater. Latest version can be downloaded from Atmel website www.atmel.com. You may download it and install it following the instructions on screen.

D2) Configure **AVR ISP** to use the 10 ways flat cable and connect it to connector CN7 of **GMM TST 2**, connect **AVR ISP** to PC serial port (please refer to instructions at points A2 and A3), configure **GMM TST 2** to program through **AVR ISP** and supply it (please refer to **GMM TST 2** manual).

D3) Run **AVR STUDIO**. AVR ISP control program can be run by pressing the button with AVR chip as icon.

D4) Select as CPU AT90CAN128, like in figure B-8.

D5) Load the file previously saved on point A4) by pressing the button "Load".

D6) Configure CPU as indicated in images of figure B-9.

D7) Configure the programmer to check signature, erase device and reprogram with verify Flash memory and configuration bits, like indicated in figure B-10.

D8) Perform the programming sequence by pressing button "Start" indicated in figure B-10.

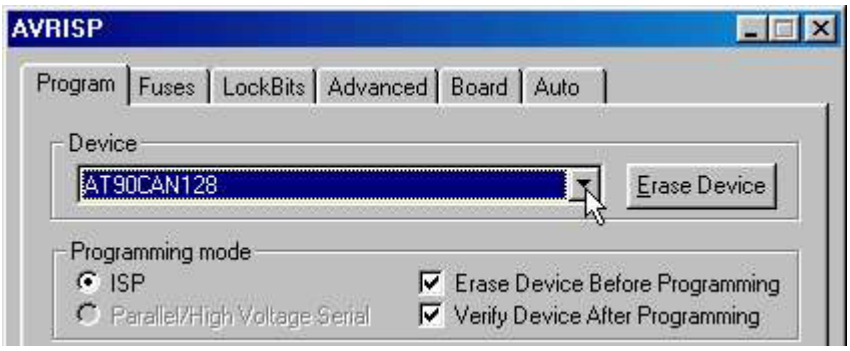


FIGURE B-8: CPU SELECTION USING AVR STUDIO

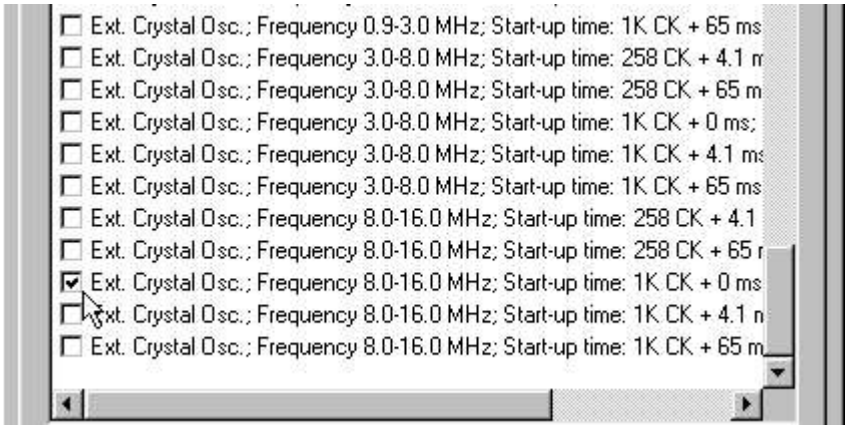

Ext. Crystal Osc.; Frequency 0.9-3.0 MHz; Start-up time: 1K CK + 65 ms
Ext. Crystal Osc.; Frequency 3.0-8.0 MHz; Start-up time: 258 CK + 4.1 m
Ext. Crystal Osc.; Frequency 3.0-8.0 MHz; Start-up time: 258 CK + 65 m
Ext. Crystal Osc.; Frequency 3.0-8.0 MHz; Start-up time: 1K CK + 0 ms;
Ext. Crystal Osc.; Frequency 3.0-8.0 MHz; Start-up time: 1K CK + 4.1 ms
Ext. Crystal Osc.; Frequency 3.0-8.0 MHz; Start-up time: 1K CK + 65 ms
Ext. Crystal Osc.; Frequency 8.0-16.0 MHz; Start-up time: 258 CK + 4.1
Ext. Crystal Osc.; Frequency 8.0-16.0 MHz; Start-up time: 258 CK + 65 r
Ext. Crystal Osc.; Frequency 8.0-16.0 MHz; Start-up time: 1K CK + 0 ms
Ext. Crystal Osc.; Frequency 8.0-16.0 MHz; Start-up time: 1K CK + 4.1 n
Ext. Crystal Osc.; Frequency 8.0-16.0 MHz; Start-up time: 1K CK + 65 m

FIGURE B-9: CPU CONFIGURATION USING AVR STUDIO

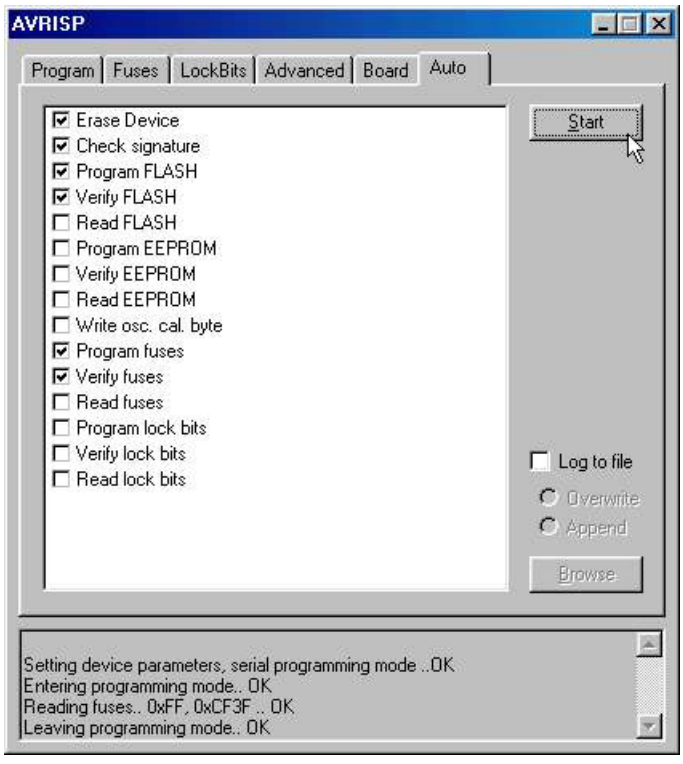


FIGURE B-10: AVR ISP CONFIGURATION USING AVR STUDIO

APPENDIX C: GMM TST2 ELECTRIC DIAGRAM

In this appendix are reported the electric diagram of **CAN GMT** support card that shows the connection modes for Mini Module signals. Detailed information on the board are available in the relative technical manual and the user can use them freely, for example to develop his own card that use the **CAN AVR** as a macro component.

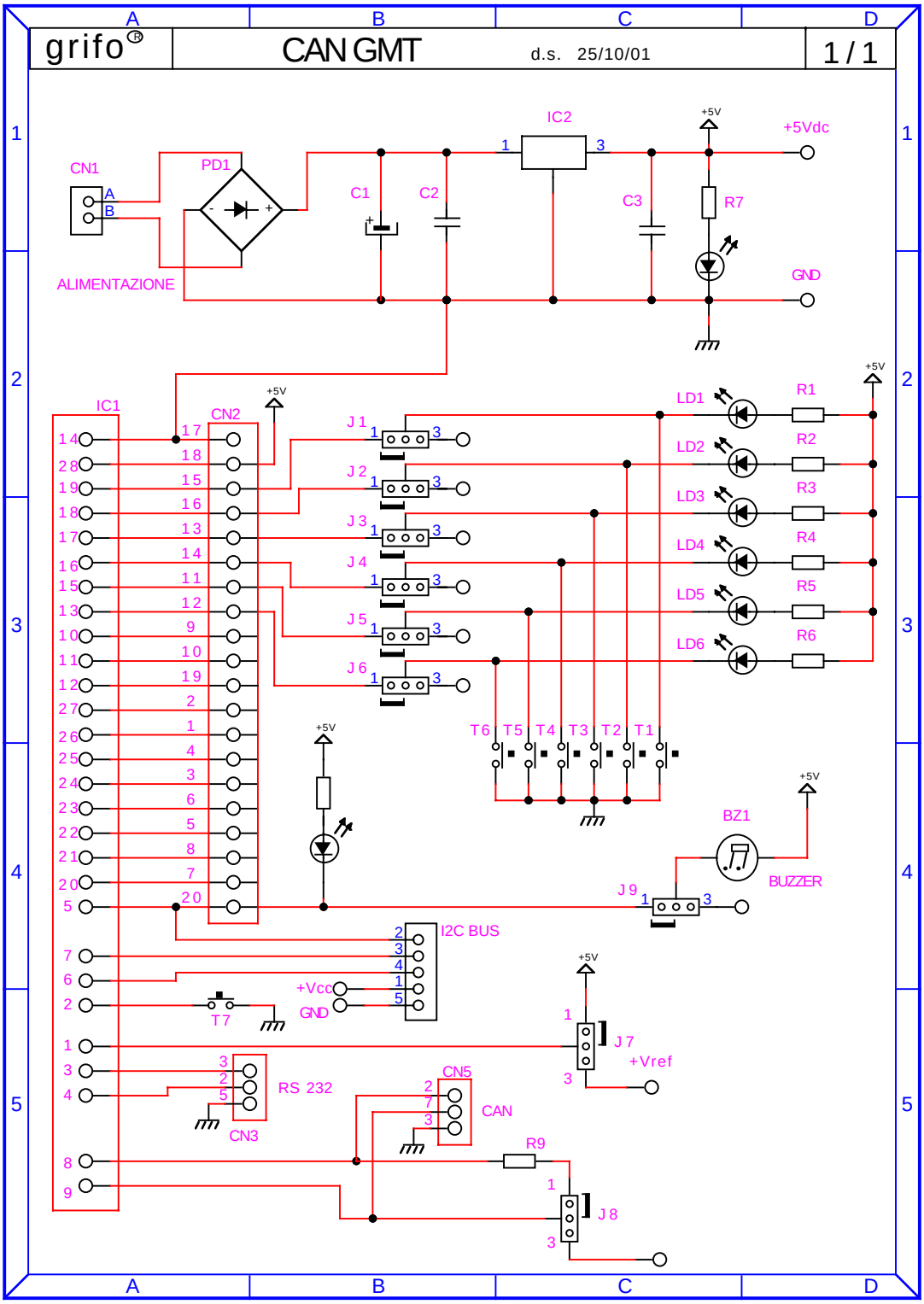


FIGURE B-1: CAN GMT ELECTRIC DIAGRAM

APPENDIX D: ALPHABETICAL INDEX

SIMBOLI

+VDC POW **13**, **15**, **20**

A

A/D CONVERTER **10**, **12**, **15**, **20**
ANALOG COMPARATOR **10**, **12**, **20**
AVR ISP **B-1**
AVR STUDIO **34**, **B-9**
AVRBOOTLOADERGRIFO **28**

B

BACK UP **9**, **10**, **16**, **37**
BACK UP BATTERY CONSUMPTION **13**
BACK UP BATTERY VOLTAGE **13**
BASCOM AVR **30**, **34**
BATTERY **9**, **10**, **16**, **37**
BIBLIOGRAPHY **38**
BOOT LOADER **28**
BROWN OUT THRESHOLD **13**

C

CAN **9**, **15**, **16**, **21**, **24**, **37**
CAN GMT **24**
CARD VERSION **3**
CLOCK **6**, **12**
CN1 **14**
CONFIGURATION BITS **28**
CONNECTIONS **14**
COUNTER MAXIMUM FREQUENCY **12**
CURRENT CONSUMPTION **13**
CURRENT LOOP **6**, **9**

D

DDS MICRO C AVR **34**
DIGITAL I/O LINES **12**
DIP SWITCH **9**, **12**, **16**, **36**, **B-2**
DSW1 **9**, **B-2**
DSW2 **9**, **B-2**

E

EEPROM **8**, **12**, **28**
EPROM **8**, **12**
EPSILON5 **B-7**

F

FLASH **8**, **12**, **28**

G

GMB HR84 **22**
GMM TST2 **B-1**

H

HOW TO START **26**

I

I²C BUS **8**, **10**, **12**, **15**, **20**
I²C BUS PULL-UP RESISTOR **13**
ICC AVR **32**, **34**
IMPEDANCE ANALOG SIGNALS GENERATORS **13**
INTERRUPT **12**, **21**, **37**
ISP **15**, **B-1**

J

JTAG **21**, **34**

L

LEDS **12**, **14**, **36**

M

MEMORY **8**
MIPS **6**
MP AVR-51+ **B-1**, **B-4**
MSI 01 **6**

P

PORTS **6**
POWER ON TIME **12**
POWER SUPPLY **13**, **15**, **20**
PROTECTION FUSES **28**
PULL-UP **8**, **37**
PWM **10**, **12**, **20**

R

REAL TIME CLOCK **10**, **12**, **37**
RELATIVE HUMIDITY **13**
REVISION NUMBER **3**
RISC **6**
RS 232 **6**, **9**, **12**, **16**, **18**, **20**
RS 232 PROTECTION **13**
RS 422 **6**, **9**
RS 485 **6**, **9**
RTC **6**, **8**, **9**, **10**, **15**, **16**, **37**

S

SCL **37**
SDA **37**
SERIAL LINES **6**
SIZE **13**
SLAVE ADDRESS **37**
SPI **8**, **12**, **20**
SRAM **8**, **9**, **10**, **12**, **16**, **37**

T

TEMPERATURE RANGE **13**
TIMER/COUNTER **10**, **12**
TTL **6**, **9**, **12**, **15**, **16**, **18**, **20**

U

UEP 49 **B-1**, **B-4**
USART **15**
USART0 **9**, **18**, **26**

W

WATCH DOG **10**, **12**
WEIGHT **13**